

| REPORT DOCUMENTATION PAGE | | | | Form Approved OMB No. 0704-0188 | |
|--|-----------------------|--------------------------------|-------------------------------|--|---|
| <p>The public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing the burden, to Department of Defense, Washington Headquarters Services, Directorate for Information Operations and Reports (0704-0188), 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302. Respondents should be aware that notwithstanding any other provision of law, no person shall be subject to any penalty for failing to comply with a collection of information if it does not display a currently valid OMB control number.</p> <p>PLEASE DO NOT RETURN YOUR FORM TO THE ABOVE ADDRESS.</p> | | | | | |
| 1. REPORT DATE (DD-MM-YYYY) 052006 | | 2. REPORT TYPE Final Report | | 3. DATES COVERED (From - To) 1 January 2003 - 30 April 2006 | |
| 4. TITLE AND SUBTITLE Achieving "Fairness" in Data Fusion Performance Evaluation Development | | | | 5a. CONTRACT NUMBER | |
| | | | | 5b. GRANT NUMBER F49620-03-1-0037 | |
| | | | | 5c. PROGRAM ELEMENT NUMBER | |
| | | | | 5d. PROJECT NUMBER | |
| 6. AUTHOR(S) Dr. James Llinas | | | | 5e. TASK NUMBER | |
| | | | | 5f. WORK UNIT NUMBER | |
| | | | | | |
| 7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Center for Multisource Information Fusion Calspan-UB Research Center, Inc. 4455 Genesee Street Buffalo NY 14225 | | | | 8. PERFORMING ORGANIZATION REPORT NUMBER | |
| 9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES) USAF/AFRL AFOSR 875 North Randolph Street Arlington VA 22203 <i>Dr John Schmusseur/NA</i> | | | | 10. SPONSOR/MONITOR'S ACRONYM(S) AFOSR AFRL-SR-AR-TR-06-0385 | |
| 12. DISTRIBUTION/AVAILABILITY STATEMENT Distribution Statement A. Approved for public release; distribution is unlimited. | | | | | |
| 13. SUPPLEMENTARY NOTES | | | | | |
| 14. ABSTRACT The key issue for and evaluation (T&E) organization is how to affordably achieve fairness in the application of its PE systems. Our PE framework provides a methodology to accomplish this; viz., the DNN Data Fusion & Resource Management (DF&RM) framework provides the hierarchical PE components for PE solution space and a methodology for mapping PE solution space into various PE problem spaces. The scope of this fairness study for performance evaluation of data fusion (DF) systems is to define a philosophy of fairness that is defensible as a basis for developing a PE system. Sample PE system MoEs need to be defined to understand the PE problem space, PE solution space and the PE problem-to-solution space mapping (i.e., the 'rules' to map the alternative PE system design solutions to the needed "Fir" PE capability). Implicitly, we are seeking design guidelines for a "best" PE that balances affordability with fairness as defined above. The first technical contribution of this report is in the reusable and extendable PE solution framework within which all applications-layer approaches to PE known to the authors can be expressed. As such, this PE framework exposes PE system design alternatives to the PE system developer and provides a common framework within which alternative PE systems can be compared. | | | | | |
| 15. SUBJECT TERMS | | | | | |
| 16. SECURITY CLASSIFICATION OF: | | | 17. LIMITATION OF ABSTRACT | 18. NUMBER OF PAGES 116 | 19a. NAME OF RESPONSIBLE PERSON |
| a. REPORT UU | b. ABSTRACT UU | c. THIS PAGE UU | | | 19b. TELEPHONE NUMBER (Include area code) |

Best Available Copy

May 2006

Achieving “Fairness” in Data Fusion Performance Evaluation Development

Satyaki Ghosh Dastidar

Center for Multisource Information Fusion, University at Buffalo, Buffalo, NY – 14260
sg44@buffalo.edu

Christopher Bowman

Data Fusion & Neural Networks, 1643 Hemlock Way, Broomfield, CO – 80020
cbowmanphd@msn.com

Introduction

In our previous work (Refs 1, 2, 3) we have been studying the definition, design, and prototyping of formal methods for the performance evaluation (PE) of Data Fusion systems. That work shows that there are in any given case alternative PE strategies that could be employed. One rationale for the down-selection of a preferred approach among such PE candidates would be to choose one which is most “fair” or equitable in comparison to the others. Since many PE cases involve comparisons among competing approaches, an equitable approach has an inherent appeal in general. Fair is defined as something “marked by impartiality and honesty” by Webster’s dictionary. In our present scope, we interpret “fair” to mean equitable and unbiased in respect to the way in which any PE approach computes the measures of performance (MoPs) and measures of effectiveness (MoEs) that are desired, based on analysis of mission-level requirements. We seek a philosophy of fairness that is defensible as a basis for conducting a performance evaluation (PE) process that yields unbiased evaluation of these MoPs and MoEs. PE systems need to be developed to maximize the probability of satisfying the PE system requirements. These can be defined by the PE system MoEs. PE Fairness for a system under test (SUT) and a selected PE MoE is achieved by maximizing the MoE for the PE process itself, which in many cases is equivalent to minimizing the error in the PE estimate of the SUT MoE (e.g., when the goal of PE is to assess the SUT operational effectiveness, regardless of the PE complexity). This can be approximated by defining MoE performance functions such as fusion track accuracy MoE thresholds, then maximizing the convolution of the MoE with the performance function (e.g., maximize the probability that the PE MoE estimate is within an error threshold). In cases where the performance function is to minimize the SUT MoE error, the PE system design can minimize the SUT MoE state error (e.g., by striving for an unbiased MoE estimate) and minimize the standard deviation of the SUT MoE error (or a user defined function of the error moments). It can happen however that different MoEs may require different PE systems to be designed. In general PE system fairness needs to consider not only fairness to the evaluation of the SUT, but also to the PE complexity and development costs (i.e., to balance PE performance and cost just as a fusion system design balances probability of mission success with complexity/cost).

The key issue for and evaluation (T&E) organization is how to affordably achieve fairness in the application of its PE systems. Our PE framework provides a methodology to accomplish this; viz., the

20061016149

DNN Data Fusion & Resource Management (DF&RM) framework provides the hierarchical PE components for PE solution space and a methodology for mapping PE solution space into various PE problem spaces. The scope of this fairness study for performance evaluation of data fusion (DF) systems is to define a philosophy of fairness that is defensible as a basis for developing a PE system. Sample PE system MoEs need to be defined to understand the PE problem space, PE solution space and the PE problem-to-solution space mapping (i.e., the 'rules' to map the alternative PE system design solutions to the needed "fair" PE capability). Implicitly, we are seeking design guidelines for a "best" PE that balances affordability with fairness as defined above.

The first technical contribution of this report is in the reusable and extendable PE solution framework within which all applications-layer approaches to PE known to the authors can be expressed. As such, this PE framework exposes PE system design alternatives to the PE system developer and provides a common framework within which alternative PE systems can be compared. That is, our PE design framework results in nominations of reasonable PE process designs to be chosen from when considering fairness criteria. The second technical contribution of the report is contained in the descriptions of "fair" yet affordable mappings of PE solutions to network centric distributed DF system PE problems.

Since perfect PE fairness as with perfect DF performance is expensive (and often unwarranted), the objective is to describe the PE engineering guidelines for achieving an optimal balance in PE "fairness versus complexity" (i.e., "knee-of-the-curve" performance). These PE engineering guidelines are driven by the PE problem space, which involves:

- DF system concept of operations (CONOPS) (mission objectives, platforms, scenarios, sensors/sources, response resources)
- DF system test articles being evaluated (DF levels, network, node functions)
- Derived DF system mission EEI metric hierarchy (MME, MSEs, MoPs)

More specifically the DF system PE problem space is organized herein as follows:

- DF mission CONOPS (mission objectives, doctrine, platforms, scenarios, constraints)
 - Mission scenarios and objectives to include: red, white, blue, grey platforms, scenario dynamics, rules of engagement, etc.
 - DF driving sources to include: on and off board sensors, IPB data bases, HUMINT, ISR platform information sharing inputs, user inputs
 - DF response resources to include: communications, countermeasures, collection, and target management plus user interfaces & responses
- DF system test articles being evaluated
 - DF fusion levels (signal, entity, relationships, COA impacts)
 - DF network (distributed over time, sources, types),
 - DF nodes (data preparation, association, state estimation)
 - DF functions (ML, MAP, MHT, JPDA, Lagrangian relaxation, unified)
- Derived DF system mission EEI metric hierarchy (measures of mission effectiveness (MMEs), measures of system effectiveness (MSEs), MoPs)

The DF system PE solution space framework is based upon interpreting PE as a fusion function where PE metrics are estimated based upon the association of SUT fusion outputs with truth or other fusion outputs (see Refs 2, 3). As such, alternative applications-layer PE solutions can be understood and described using the data fusion portion of the data fusion and resource management (DF&RM) dual

node network (DNN) technical architecture. Thus the PE solution space can be organized as a network of PE functional nodes where each PE node performs fusion track and truth data preparation, data association, and MoP state estimation. The PE network and node design guidelines strive to achieve the knee-of-the-curve in PE 'fairness' and complexity.

The PE design objective is to generate fair DF system mission EEI metrics (e.g., MME, MSEs, MoPs) with minimal cost. As such the PE MoPs need to be computed with sufficient accuracy to differentiate SUT performance with respect to the scenario MoPs which are driven by the mission objectives, scenarios, and the SUTs. Thus, engineering guidelines for fair PE are driven more by the scenario and their MoPs and less by whether the SUT is simple or sophisticated (e.g., deterministic, probabilistic, or unified association and fusion). For example, a sophisticated SUT may be needed to associate the sensor reports whereas a simple PE may be sufficient to associate the resultant scenario tracks-to-truth (although this is not expected). That is, we do not believe there is a one-to-one mapping between the degree of sophistication in SUT fusion process design and PE fusion process design. Scenarios and their MoPs are selected based upon the mission needs and not upon the SUT limitations. However, when competing SUTs are expected to perform equivalently on most of the mission needs, the PE may be focused on the distinguishing scenarios and their MoPs.

The report is organized as follows. In Section 1, we describe the PE problem space. In Section 2 we discuss the PE solution space in great detail and finally in Section 3, we describe the candidate PE network and node design guidelines.

1. PE Problem Space

Traditionally data fusion system performance is evaluated in terms of the probabilities of satisfying the needs for accuracy, completeness, and timeliness of Essential Elements of Information (EEIs) to support selected missions. The EEI's are derived based upon a flow-down from the mission operational needs. Sensors and off-board sources provide reports that enable the fused estimate of the identity, location, track, aggregations, activity relationships, capability, intent, courses of action, and impact for entities of interest with corresponding errors, deficiencies, and latencies. The DF entities of interest in fighter mission applications include such elements as aircraft, ships, vehicles, ground sites, individual terrorists, battery, company-level, and higher level units with corresponding EEIs. Performance evaluations can be for a wide variety of mission applications and extend beyond data fusion into resource tasking (e.g., data collection, targeting, countermeasures, etc.) to support decisions to invoke avoid, evade, deceive, degrade, and kill operations by diverse assets. The focus herein will be for the PE of the contribution of USAF tactical aircraft distributed data fusion avionics to support EEI satisfaction for tactical missions. The following sections describe the 3 partitions of the PE problem space defined above.

1.1 DF Mission CONOPS PE Problem Space Drivers

The DF system role is defined by the battlespace missions, doctrines, platforms, and scenarios that need to be accomplished. The missions that the DF system needs to support include the mission objectives and the topology of the physical space where the action is taking place, the physical laws, the involved equipment and the entities' physical attributes. The doctrine includes the rules of engagement and policies.

The Data Fusion (DF) system role PE problem space drivers can be decomposed into 2 main components described in the following subsections.

1.1.1 DF Mission Scenarios and Objectives

The CONOPS with the resulting fusion system design criteria and constraints define the problem for the fusion system design to be evaluated. The CONOPS for multiple fighters requires netted fusion processes (e.g., real time prosecution of ground and air targets, mixed conventional/Low Observable (LO)/UAV operations, real-time bomb damage assessment (BDA), joint operations). This requires integration of collaborative situation assessment and precise targeting with real-time planning and coordination of collection management assets. The result is cooperative network-oriented air combat.

The ownship, cooperative, and off-board distributed L1, 2, and 3 fusion test articles are driven by a variety of mission scenarios. Each scenario vignette is defined within a joint operations mission context. Missions include:

- Interdiction for non-emitting targets (e.g., airfield complex) and Theater Air Defense (e.g., SCUD TEL)
- CAS non-emitting targets (e.g., armor)
- Electronic Combat (EC) -Lethal SEAD for mobile emitting targets (e.g., SAM radar)
- High value asset protection: AWACS, JSTARS, Compass Call, ABCCC, Rivet Joint, E2C, EP-3E, UAVs, Tankers

Projected fighter mission scenario sequencing include:

1. Establish air superiority
2. Eliminate long range threats
3. Jam enemy radars
4. SEAD
5. Destroy airfields
6. Escort strike forces for CAS and TAD

Sample scenario assumptions include:

1. Use of medium altitude profiles to negate AAA and optimize stand-off munitions
2. Inflight data link (IFDL) high speed data link is available with sub-second latency.
3. A/G weapons include: JDAM, AGM-130, and GBU-15 with terminal seeker and JSOW with enhanced GPS.
4. A/A weapons include: AMRAAM, AIM-9X
5. Integrated Navigation with GPS

Sample scenario vignettes include:

1. A/G or A/A Distributed fusion and management to achieve orthogonal passive simultaneous target tracking preceded by surveillance driven by IPB targets for 4 fighters under EMCON with 2 standoff jammers and 2 CAP.
2. A/G or A/A Distributed fusion and management to shorten time to achieve high confidence ID on fused tracks.

3. A/G or A/A Distributed fusion and management for improved large volume surveillance search especially in front ± 30 degrees A/A. Coordinated surveillance search combined with prioritized updating of existing targets and tracks of interest.
4. Detect & ID SAM launches using coordinated tactics to improve time to ID, range, threat prediction. Minimize exposure especially of aircraft not launched upon.

The distributed Data Fusion (DF) objectives tailored for each mission are include:

- maximize situation awareness quality
 - improve system detection performance
 - reduce system false alarm rate
 - minimize number of false/redundant tracks
 - increase probability of correct ID
 - extend situation awareness range
- improve target acquisition time and accuracy for weapons quality solution
 - facilitate target handoff from other platforms (e.g. C2)
 - facilitate sensor-target cueing (e.g. sensor handoff)
 - minimize target location and kinematics errors
 - improve combat ID performance
- improve survivability
 - decrease time required for target/threat detection
 - decrease time required for target/threat localization
 - decrease time required for target/threat ID declaration
- reduce sensitivity to individual sensor faults

The DF design constraints are summarized as follows:

- to minimize risk
 - avoiding “nothing works until it all works”
 - incremental build plan
- cost versus performance
 - modular and maintainable
 - existing best algorithms to be exploited
 - extendable to expand capability over lifecycle
 - versatility to use minimal resources to provide necessary information at the appropriate time
- operational considerations
 - decouple sensor management generation of system information needs from fusion processes by eliminating the need for fusion to generate sensor requests
 - ability to display individual sensor, ownship, intra-flight, and total composite track information

1.1.2 The DF Driving Sources

Typical sensors and avionics available on each fighter include:

- Radar Multi-Function Array (MFA): air-to-air all aspect search, multi-target track, high gain ESM with passive ranging, autonomous search, cued search, weapon data links, and A/G modes including SAR map and GMTI.

- EW: multiple band radar warning and forward air-to-surface with precision direction finding (PDF), emitter ID, distributed passive ranging, and ECM
- CNI: IFF, intra-flight data link MADL, SATCOM intelligence broadcast links, voice and data communications, messaging, GPS, integrated navigation, IRS, TACAN and other landing aids
- Integrated Electro-Optical Tracking System (EOTS): FLIR (air-to-ground), laser ranger, laser designator,IRST (air-to-air), missile launch detection (MLD), and laser spot tracker
- Distributed A/A and A/G Fusion: cooperative and off-board broadcast data fusion track files,
- Aircraft Status: avionics status and environment data
- Intelligence Preparation of the Battlespace (IPB): threat laydown, weather, EW data base, etc.

Key sensor driving parameters for each sensor mode are the field of regard, field of view, update rate, at reference range probability of detection per target type (e.g., 90% detection range), false alarm rate, (*R Az El*) accuracy and resolution, mode switch delay, beam slew rate, IFF/class/type ID classification accuracy (i.e., confusion matrices) per range and aspect bin, type, and environment condition.

1.1.3 The Fighter Response Resources

Typical avionics resources that use the outputs of fusion on each fighter include:

- mission management (power, propulsion, diagnostic),
- pilot/crew controls and displays (HUD, HMD/HMS, HOTAS, MFD, audio, controls)
- information management (sensors, CNI, fusion process),
- adjudication management (maintain consistent tactical picture across platforms)
- communications management (balance BW with mission communications needs)
- threat and target response management (flight, weapons, countermeasures)
- stores management: stores control and stores interface

Distributed fusion output needs are driven by these response system needs to meet mission objectives.

1.2 DF Test Article PE Problem Space Drivers

PE system design is also driven by the DF SUT to be evaluated. The types of DF SUTs can be organized using the DNN DF&RM architecture as follows:

- DF fusion levels
- DF network distributed over data fusion levels 0 through 3 (i.e., signal, entity, relationships, COA impacts), time, sources, and data types
- DF nodes performing data preparation, data association, and state estimation)
- DF functions in each fusion node (e.g., ML, MAP, MHT, Lagrangian relaxation, track filters, ID combination, JPDA state estimation, and unified (e.g., random set) entity state combination)

Having the DNN framework within which to understand and compare the alternative distributed fusion SUTs eases the development and expression of the engineering guidelines for PE system design based upon this portion of the PE problem space.

1.3 DF System Mission Metric Hierarchy PE Problem Space Drivers

Since there are so many EEIs at different aggregations and echelons of interest, it is practical to organize the EEIs hierarchically with the highest rank being those that define the measures of mission effectiveness (MME) as depicted in Figure 1-1. Since DF is usually only a contributor to overall mission success, a comparison of alternative DF systems usually entails a fixing of all the DF source and response capabilities while alternative DF systems are assessed. Other PE systems can be developed to evaluate alternative sources or response resources given a fixed DF. For DF systems that must operate over many missions and scenarios per mission these scenario level metrics can be combined in many ways (e.g., weighted sums, concurrent requirements, ad hoc).

Since the highest performance metric (i.e., the MME) is too coarse for many DF evaluations and comparisons, the DF PE system will need to be tailored to compute the appropriate levels in the hierarchy of EEIs based upon the DF mission CONOPS and the alternative DF SUTs being evaluated. The tailored DF system mission EEI metric hierarchy (MME, MSEs, MoPs) is derived based upon these primary PE problem space drivers.

In a data fusion network design, the key metric types include (1) probability, (2) error, and (3) information. Probability is a normalized ratio of performance over a complete set of possibilities. Error is associated with uncertainty. Uncertainty is the result of the randomness of situational constraints that result from Fusion system performance in real-world testing. Such an example of uncertainty is the typically unpredictable latency associated with incoming data. Information is a quality metric associated with the value of the data fusion to meet functional needs. To develop any metric for system level performance, we use probabilities (P), errors (σ), or time (t), as contributing to the system metrics. Information metrics, such as mutual information or entropy describe the fusion information

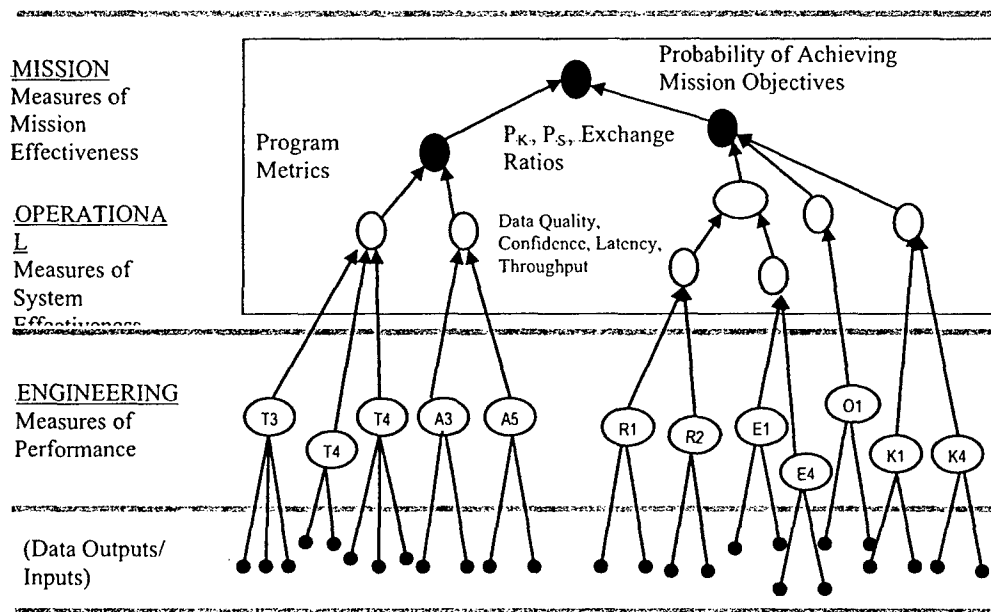


Figure 1-1: Hierarchy of EEI Metrics for Each Data Fusion System (Steinberg)

gain [Refs 4,5,6]. Error and probability relate to confidence and accuracy.

Metrics determine fusion performance and can include (1) an Objective (desired) and (2) Threshold (minimum acceptable requirement). If we look at the fusion system from top down (satisfy user needs) or bottom-up (minimize uncertainty), the goal is to define the metrics for the evaluation of fusion systems in support of the avionics mission. Examples of metrics for each fusion level are as follows:

| | | |
|-------------------------------------|---|---|
| Level 0: Sub-Object Data Assessment | – | Positional error |
| Level 1: Object Assessment | – | Probability of tracking and ID accuracy |
| Level 2: Situation Assessment | – | Relationship accuracy, |
| Level 3: Impact Assessment | – | Survivability, Vulnerability |

As described above the selection of the fusion MoPs are derived from the MMEs. The MoPs support the MoEs by providing specific performance insights. Examples of MoPs for distributed fusion include:

- *CTP Consistency* is the average percentage of non-matching CTP tracks after a suitable time communications delay (2 sec)
- *CTP Update Time Delay* is the average over all sensor inputs of the difference in the input time until the update is generated both locally and globally on each platform.
- *CTP Association Accuracy* is the percentage of correct associations per track over time averaged over CTP tracks to yield overall association accuracy per scenario (a lower level MOP).
- *BW Utilization* is defined as the peak and average percentage of communications BW load for each scenario

Examples of single platform level 1 fusion metrics include:

- *CTP Kinematics Accuracy* is defined as the standard deviation over time of the error in the CTP averaged over all tracks for each scenario. Averages Over all platforms and scenarios for all MoPs will also be taken for further condensation of the performance.
- *CTP Classification Accuracy* is the percentage of CTP tracks with conflicting classification averaged over time for each scenario
- *CTP Track File Probability of Detection* is defined as the number of track to truth associations at that time divided by the total number of truths existing at that time. These are then averaged over time for each scenario.
- *CTP Probability of False Track* is be computed as the number of CTP tracks not associated with any truth at that time divided by the number of CTP tracks at that time. These probabilities are then averaged over time for each scenario.
- *Computational Complexity* includes processing timing and sizing

Refinements of level 1 fusion metrics are used to provide additional insights into their corresponding MoPs. Association MoP refinements include:

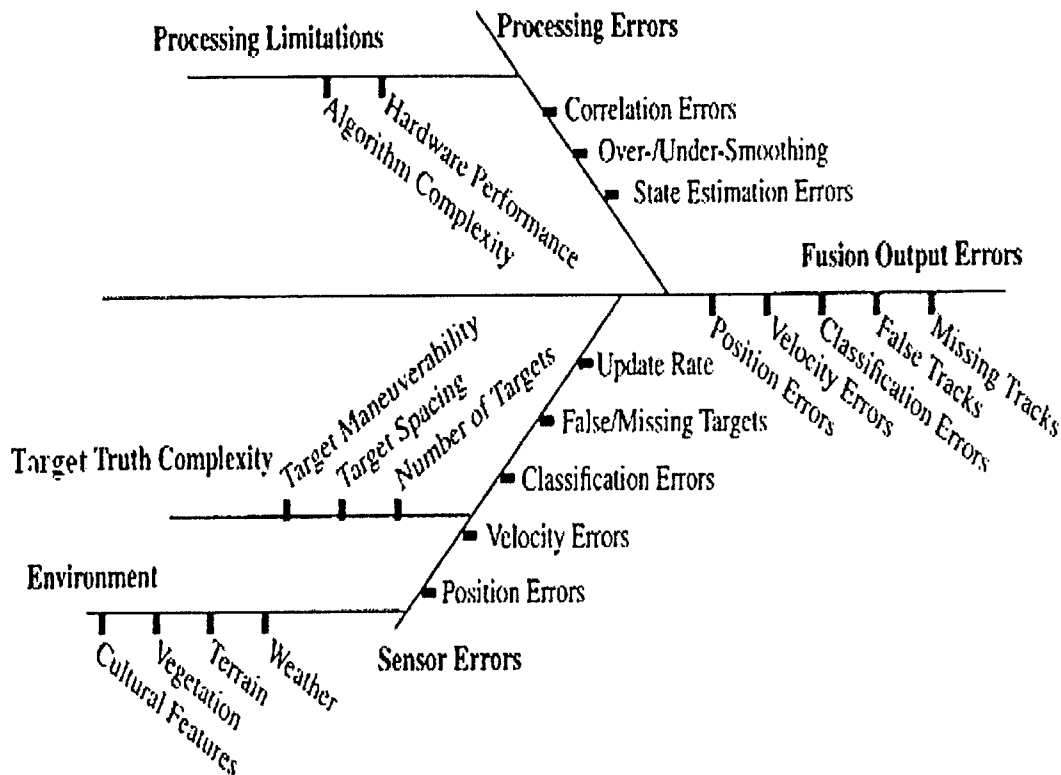


Figure 1-2: Contributing Factors for Fusion System Output Performance [Ref X XX10].

- *Track Purity (Targets/Track)*: the correlation coefficient of the pairing of elements which the system assigns to a hypothesized aggregation and elements of the corresponding ground truth entity
- *Track Fragmentation (Tracks/Target)*: the number of hypotheses to which elements of an actual aggregation are assigned as elements by the system
- *Hypothesis Proliferation (Tracks/Report)*: the number of competing (overlapping) tracks per report
- *Assignable Track Ratio*: fraction of tracks that are associated with exactly one target
- *Non-Assigned Target Ratio*: fraction of targets to which no tracks are assignable

This rest of this Section provides examples of fusion MoPs. Figure 1-2 shows the Cause and Effect diagram for the Factors influencing the PE Problem Space (from Ref 7). The Performance of a system is not only affected by System Parameters, i.e. “Solution-space” or “Design-space” independent variables, but also by Scenario Context variables, those mostly on the lower-half of Figure 1-2, such as Track Truth Complexity and Environmental Factors; these variables affect DF performance in many ways [Refs 8, 9].

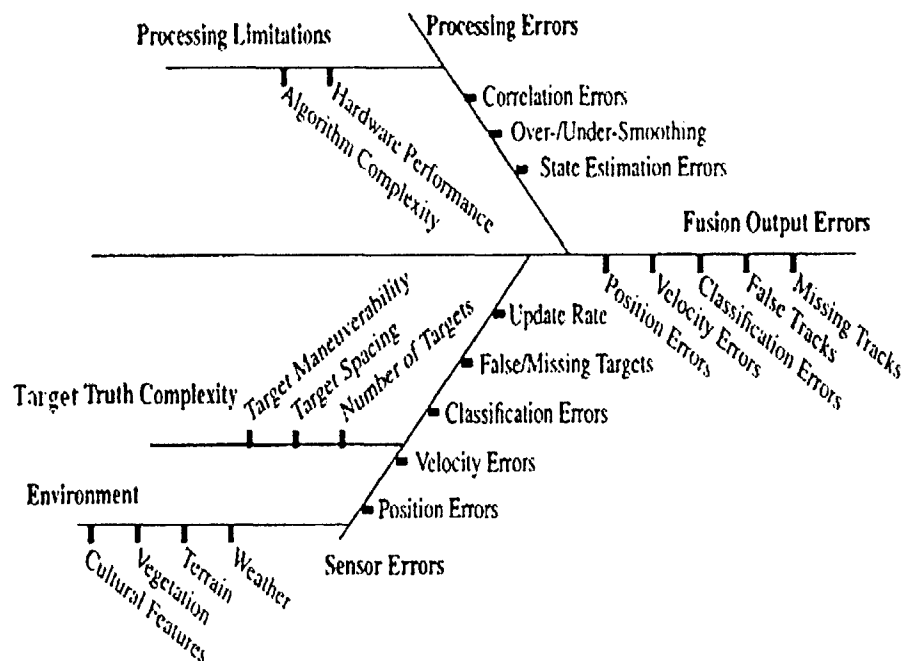


Fig. 1-2 Complicating Factors in Tracking and Fusion Performance Evaluation [Ref 9]

Some of these are:

- **Target Truth Complexity:** Variables such as maneuverability challenge the tracking algorithm's ability to handle targets that suddenly change direction; target spacing affects the design and performance of tracking gates and data association; numbers of targets challenge the computational efficiency of a tracking algorithm
- **Environment:** Independent variables such as weather, which affect both target abilities and sensor abilities, and terrain and vegetation, which affect a ground target's maneuverability, can have a wide range of specification, as each of these variables can have sub-variables in turn that, in combination, affect the status of a variable such as weather (i.e. weather = {temperature, humidity, precipitation (nature and degree), wind conditions (direction, magnitude), etc.}).
- **Time Asynchronicity:** This problem happens when two sensors (platforms) are asynchronous in time and the track reports from each of them are not synchronized, thereby resulting in track association problems.
- **Geo-Location Error:** The geo-location error is defined as:

$$\epsilon_{\text{Geo-error}} = \sqrt{\frac{1}{N_c} \sum \epsilon_{\text{Lat}}^2 + \epsilon_{\text{Long}}^2} \text{ (meters)}$$

- where: N_c = Number of control points,

- $\varepsilon_{Lat} = (\text{Latitude Reported} - \text{Latitude Truth})\alpha_{Latitude}$,
- $\varepsilon_{Long} = (\text{Longitude Reported} - \text{Longitude Truth})\alpha_{Longitude}$,
- Conversion: $\alpha_{Latitude} = \text{Latitude Error (WGS-84) to meters}$,
- $\alpha_{Longitude} = \text{Longitude Error (WGS-84) to meters}$.
- **Accuracy modeling** – KS Statistic, Chi-Square Test, or Wald Test: The Kolmogorov Smirnov (Goodness of Fit) test statistic is defined as:
 - $$D = \max_{1 \leq i \leq n} \left| F(Y_i) - \frac{i}{N} \right|$$
 - where F is the theoretical cumulative distribution being tested, Y_i are the ordered set of points from 1 to N , and D is the statistic compared to a table (based on sample size N) to determine if the observed registration is within the truth registration distribution.
- Factors or independent variables can also be related to the “Solution Space”, meaning the Factors that affect the performance of particular fusion algorithms (e.g., the nature and number of models in an Interacting Multiple Model tracker). Thirdly, and peculiar to the nature of the overall PE process, there are Factors involved in the Performance Evaluation approach itself, such as the choice of technique for Track-to-Truth assignment, or the Factors upon which a specific PE Tree might be partitioned [Refs 2, 10]. There are *three classes of independent variables or Factors* around which the PE process revolves: Problem-space Factors, Solution-space Factors, and PE process Factors. We analyze the various PE Solution Space factors based on the Dual Node Network (DNN) architecture.
- **Detection, ATR, and Identification Metrics: (Level 1)**
- Target information can be modeled as per the NIIRS rating: detection, recognition, classification, and identification.
- **Probability of Detection (P_D)** – The ratio of the number of recorded detections (N_D) to the number of detection opportunities (N_{DO}). ($P_D = N_D / N_{DO}$).
- Note: P_D is applicable to stationary and moving targets, where emitters can be inferred as detections. A moving target is said to be detected if a set of reports corresponding to the target are associated and a vehicle track is declared. A stationary target group is said to be detected if more than $X\%$ of the targets within the group are detected and associated with one another, where X is a parameter. A moving target group is said to be detected if a set of reports corresponding to more than $Y\%$ of the targets comprising the group are associated and a group track is declared, where Y is a parameter.
- **Probability of Recognition (P_R)** – Ratio of correct type declarations (N_R) to opportunities (N_{RO}). ($P_R = N_R / N_{RO}$)
- **Probability of Classification using Confusion Matrices** – A common reporting format for ATR systems is classification probabilities, including cross target probabilities, associated with a given population. For example, a Confusion matrix records the probabilities $P(i | j) = \text{probability of declaring a target as type } i \text{ given that it is really of type } j, \text{ where } i \neq j$,

- An algorithm for Probabilistic Multiple Hypothesis Testing with classification measurements (PMHT-c) is presented by Davey [Ref 11]. The algorithm was derived under both known (or assumed) classifier statistics, and unknown classifier statistics. When the classifier confusion matrix is unknown, the PMHT-c can estimate it. The PMHT-c was shown to simplify to the standard PMHT when the classification measurements were known to be uninformative, and to revert to hard association when the classification measurements are known to be perfect.
- **Probability of Correct Identification (P_{ID})** – Ratio of the number of times a target, emitter, or group is correctly identified (N_{ID}) to the number of occurrences (N_{IDO}). ($P_{ID} = N_{ID} / N_{IDO}$).
- For example, we can use shape metrics for ID evaluation, e.g., RMS errors on length and width target attributes:

$$RMS_{Length} = \sqrt{\frac{1}{N} \sum [Length_{True} - Length_{Estimated}]^2}; RMS_{Width} = \sqrt{\frac{1}{N} \sum [Width_{True} - Width_{Estimated}]^2}$$

- Other metrics include a Log-likelihood ATR, Maximum *A Posteriori* (MAP), and maximum likelihood (ML) [Ref 12]. The ML is based on the measurement information while the MAP is based on the expectation from the filtering analysis. As described by the Kalman Filter, we see that ML is used in the association. Estimation and prediction filtering use the MAP which is achieved from a Bayes analysis. In determining the true target analysis, we also desire to determine the error of the analysis using a false alarm metric:
 - **False Alarm Rate** – Number of false detections per square kilometers (km^{-2}).
 - **Track Metrics: (Level 1)**
 - **Probability of Track Detection (P_{DT})** – Ratio of detected tracks (N_{DT}) to true track number (N_{TT}). ($P_{DT} = N_{DT} / N_{TT}$).
 - **Track False Alarm Fraction (F_T)** – Ratio of false tracks (N_{FT}) to total tracks (N_{TT}). ($F_{FT} = N_{FT} / N_{TT}$).
 - **Track Continuity** – Average number of tracks formed per trajectory of a single target. Ideally equal to 1.
 - **Track Purity (T_P)** – Ratio of track segments in an integrated track that belongs to same target (or group of targets (N_{TS}), to total number of segments in a track (N_{TST}). ($T_P = N_{TS} / N_{TST}$).
 - **Track Position Accuracy** – Root Mean Square Error between ground truth and tracker target positional estimates:
- $$RMSE_{TPA}(N) = \sqrt{\frac{1}{N} \sum_{i=0}^{N-1} [(x_i - \alpha_i)^2 + (y_i - \beta_i)^2]} \text{ (meters)}$$
- where, x_i, y_i are sensor estimates of target positions, α_i, β_i are true target positions, and N is a specified number of samples defining the observation period.

- **Track Heading Accuracy** – RMS Error between true target heading and sensor estimates of target heading:

$$RMSE_{THA}(N) = \sqrt{\frac{1}{N} \sum_{i=0}^{N-1} (\theta_i - \phi_i)^2} \text{ (degrees)}$$

- where θ_i is sensor estimates of target heading with respect to North, ϕ_i is true target heading with respect to North, and N is a specified number of samples.

- **Track Velocity Estimate Accuracy** – RMS Error between truth velocities of targets and sensor velocity estimates of targets:

$$RMSE_{TVA}(N) = \sqrt{\frac{1}{N} \sum_{i=0}^{N-1} (v_i - V_i)^2} \text{ (meters/sec)}$$

- where v_i are sensor velocity estimates of targets, V_i are target truth velocities, and N is a specified number of samples.

- **Target Flow Rate Accuracy** – Root Mean Square Error between estimated target flow rate and truth target flow rate:

$$RMSE_{FE}(N) = \sqrt{\frac{1}{N} \sum_{i=0}^{N-1} (\xi_i - \gamma_i)^2} \text{ (degrees)}.$$

- where ξ_i is estimated target flow rate (targets/sec), γ_i is actual target flow rate, and N is a specified number of samples.

- **Situation (Relationship) Assessment Metrics (Level 2):**

- Situation awareness or assessment is typically evaluated based on mission needs. At higher levels of fusion, the lower level metrics can be aggregated. For instance, individual entity metrics of accuracy can be aggregated for group metrics such as group spacing, group identity, and relational aspects of group members (how likely are they to be members of the same group). Situational metrics are derived from user needs for situational awareness. Metrics include: attention, workload, trust, and dependability [Ref 13]. Attention and workload correlate to the communications *throughput* of the information. While lots of data could be time-consuming, it is assumed that the fusion system would deliver a parsimonious, reliable set of results to the user. Trust is related to *confidence* in presented results. Finally, dependability is related to *cost* since the situational content can either take time away from the user (opportunity cost) or minimize the effort needed to explore alternatives. As an example of situational metrics, we suggest relationship association (matching of level one entity attributes of tracking and identification to relationships) metrics:

- Probability of Correct Association (P_{CA}) – Ratio of the number of correct needed relationship correspondences in L2 fusion outputs to the number of truth relationships needed to complete mission.
- ($P_{CA} = N_{CA} / N_{CAO}$).
- Probability of relationship detection (P_{TCA}) – Ratio of the number of correct track entity correspondences in L2 fusion outputs, (N_{TC}), to number of truth relationships needed
- (N_{TCO}). ($P_{TCA} = N_{TC} / N_{TCO}$).

- Percentage of False Relationship Declarations – number of false relationship declarations divided by the number of truth plus false relationship declarations
- Accuracy of Relationship State Update Declaration –percentage error in the needed relationship declaration reported confidence at or before time needed to complete mission averaged over all relationships needed.
- **Impact Assessment Metrics (Level 3):**
- Impact assessment relates to benefits, costs, and risks. Since a fusion system is employed to reduce uncertainty, maximize information, or maximize probability of mission success, it is important to choose metrics that address tradeoffs as a function of risk of mission failure. For example, typical risk metrics include:
 - Aircraft Survivability – probability that the platform survives the mission (e.g., jeopardy from threat) and
 - Target Vulnerability – vulnerability of prospective target to own ship aggressive action.
- Four typical L3 fusion metrics are:
 - Percentage of correct coarse of action prediction over all threats in time for defensive action -- # correct COA predictions/total number of COA predictions needed
 - Percentage of correct coarse of action prediction over all targets in time for offensive action -- # correct COA predictions/total number of COA predictions needed
 - Accuracy of the probability of survival, P_s , in time for defensive action: $[P_s(\text{computed}) - P_s(\text{truth})] / P_s(\text{truth})$
 - Accuracy of the probability of vulnerability, P_v , in time for offensive action : $[P_v(\text{computed}) - P_v(\text{truth})] / P_v(\text{truth})$
 - Another Level 3 metric could be an exponential time decay (based on the *a priori* information) on the confidence of information generated over time. The longer the delay means the higher the uncertainty in mission completion and the greater the risk.
- To measure interactions between future fusion system designs and users needs, additional metrics are required. Blasch *et al.* [Ref 14] discuss a set of fusion metrics to bridge the user-fusion gap. The metrics chosen include timeliness, accuracy, throughput, confidence, and cost. These metrics are similar to the standard QOS metrics in communication theory and human factors literature, as shown in Table 1-1.

| Communication | Human Factors | Info Fusion | ATR/ID | Track |
|-------------------------|---------------|--------------|----------------------------|-------------------------|
| Delay | Reaction Time | Timeliness | Acquisition/ Run Time | Update Rate |
| Probability of Error | Confidence | Confidence | Prob. (Hit), Prob. (FA) | Prob. Of Detection |
| Delay Variation | Attention | Accuracy | Positional Accuracy | Covariance |
| Throughput | Workload | Throughput | No. of Images | No. of Targets |
| Cost | Cost | Cost | Collection platforms | No. of Assets |
| Stallings 2002 | Wickens, 1992 | Blasch, 2003 | Blasch, 1999 | Blasch, Hoffman 2000 |

Table 1-1: Traditional Metrics for Various Disciplines

2. Performance Evaluation Solution Space

DF system PE solution space framework is based upon interpreting PE as a fusion function where PE metrics are estimated based upon the association of fusion outputs with truth or other fusion outputs. As such alternative applications-layer PE solutions can be described using the data fusion portion of the Data Fusion and Resource Management (DF&RM) Dual Node Network (DNN) technical architecture. Thus the PE solution space is organized as a network of PE functional nodes where each PE node performs fusion and truth data preparation, data association, and MoP state estimation. The PE network and node design guidelines strive to achieve the knee-of-the-curve in PE ‘fairness’ and complexity. Descriptions of candidate PE network and node framework are organized in this Section as follows. Section 2.1 describes the Performance Evaluation System Role Optimization; and Section 2.2 outlines the methods for Performance Evaluation Fusion Network Optimization. Section 2.3 described the PE Fusion Node functional components.

2.1 Performance Evaluation System Role Optimization

The first step in the PE system development process is to define the role for the performance evaluation software based upon the scenario performance evaluation requirements.

2.1.1 PE System Concept of Operations (CONOPS)

Performance Evaluation of blue and red (friendly and hostile) distributed DF&RM is performed within the AFFTC Test and Evaluation (T&E) as depicted in Figure 2-1. For this research PE will be performed within the CMIF Distributed Fusion Performance Evaluation (DFPE) testbed described in our CY04 final report. The CONOPS within the AFFTC remains to be determined based upon their distributed fusion environment.

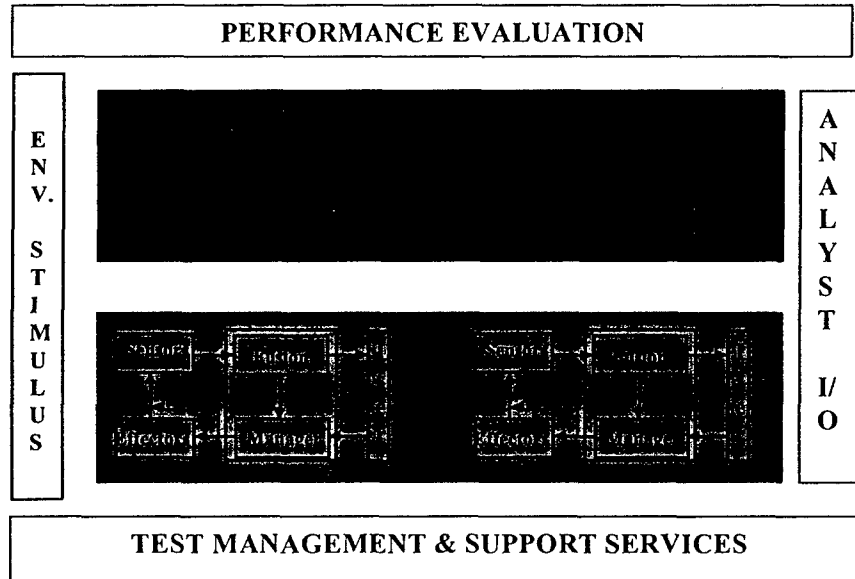


Figure 2-1: Performance Evaluation Operates within the AFFTC T&E Framework

The PE measure of mission effectiveness (MME) for this research is the fairness of the PE result achieved within the budget for this effort.

2.1.2 PE Black Box Design

The baseline black box role for Performance Evaluation is shown in Figure 2-2. Namely, PE receives the tactical picture and response commands output from the test article fusion nodes. PE receives the best estimate of truth from the scenario generation stimulus. PE receives sensor status from the sensor models and other scenario related information from support services. PE outputs its results and receives commands from the analyst. The PE system can be developed using the DNN architecture to specify a network of PE nodes. Each PE node estimates the fused track file MoPs based upon the association of the Consistent Tactical Picture (CTP) tracks with truth or other fusion tracks.

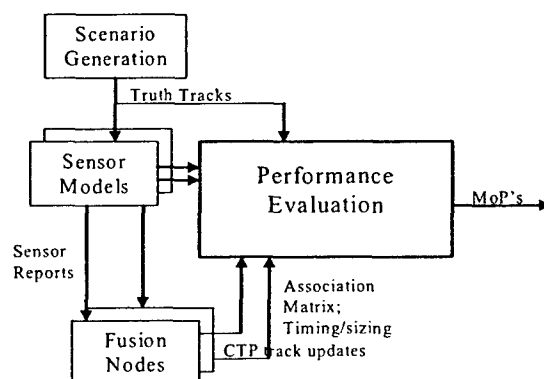


Figure 2-2: Sample Performance Evaluation Associates the Distributed Fusion System CTP Tracks with Truth Tracks to Generate MoPs

2.1.3 PE System Role Optimization

The PE system role evaluation for role optimization is based upon the PE MME and measures of effectiveness (MoE) and their derived MoPs as described above. The following sections describe how the PE network and nodes are designed to achieve the knee-of-the-curve in PE system performance versus cost based upon further breakdown of the MoEs. In general, PE role optimization criteria include

- PE fairness (e.g., accuracy especially for those MOPs that distinguish the envisioned SUTs)
- PE system cost & complexity (e.g., usually stipulated as a constraint budget)

2.2 Performance Evaluation Fusion Network Optimization

2.2.1 PE Requirements Refinement

Specific PE MoPs will be defined after the distributed fusion system under test (SUT), its MME/MoE/MoPs, and scenario environment has been determined. Other quality factors such as reliability, redundancy, maintainability, availability, portability, flexibility, integrity, reserve capacity, robustness, etc. will be considered later.

2.2.2 PE Fusion Network Design

The PE MoP estimation requires the association of the DF SUT Consistency Tactical Picture (CTP) output tracks to the truth or other fusion SUT track entity states. To simplify this discussion the track-to-truth solution approaches will be described in remainder of Section 2 herein. The track-to-track extensions are similar.

A trade-off of performance versus complexity must be made to design the track-to-truth association and resulting MoP estimation software. This Section describes how the CTP track data is to be batched (e.g., over time, scenarios, platforms, sensors, reports, etc.) to be processed by PE nodes. Section 2.3 then specifies how the scoring for each PE fusion node for association with truth is to be accomplished (e.g., track-to-truth association scoring).

The best performing approach is to take all the data from a scenario in one large batch and then perform an 'optimal' (retrospective) estimation of the MoPs with appropriate consideration of the association of all the information over all time. Note however that there is still the choice of PE network design that influences the results for any batching strategy. To reduce the cost and complexity of the association part of such an approach significantly, the traditional approach is to associate each CTP track to the truth entity that originated 'most' of its associated reports over the scenario. The flaws in this solution include the following:

- two CTP tracks can have 'most' of their associated reports from the same truth track,
- many accurately associated reports may not compensate for a few inaccurately associated reports,
- forced associations are not indicative of the true performance (e.g., for crossing entities where the only ambiguity is at the cross, the CTP track may switch truth entities at that point and then have what appears to be many incorrect associations thereafter).

- flipped entity associations due to close tracks that do not have ‘most’ of their associated reports from any one truth entity, and
- PE association solution relies on the association accuracy of the distributed fusion system nodes that it is trying to evaluate.

Note that one could also select to ‘not decide’ how to associate the CTP tracks to truth for ‘hard to call’ ambiguous cases, thus necessitating another MoP (e.g., ambiguity percentage).

A driving requirement on the PE design is to minimize cost/complexity. In contrast to a batching approach, the simplest PE tree is one report at a time; however it has the least accurate association. One simple solution for this tree is to associate the CTP track update with the truth from which the associated report was generated. However, this solution, as for the one above, relies on the association accuracy of the fusion nodes. Another approach is to score and select the CTP to truth association based upon the updated CTP and truth states every time the CTP is updated. This has the benefit that the PE tree is batched identically as the fusion tree is batched. The fundamental flaw with these one update at a time fusion trees is that the same CTP track could associate to many different truths over time and vice versa. Thus, these simple PE trees would not detect that there may be more or fewer CTP tracks than truths.

What really counts in meeting the requirements is to measure in the PE process how accurately the CTP reflects the truth. To achieve that with minimum cost, the baseline PE fusion tree will associate the current CTP to truth at selected time points and then estimate the MoPs using this association at the necessary time points (e.g., at updates), as described in Figure 2-3. In this approach, the whole CTP is associated with the complete truth states using all available information (e.g., kinematics, attributes, ID/type). However, there are sub-optimality problems with this batching of the track-to-truth association. These include any situation where taking a larger batch of data would remove ambiguities (e.g., using an MHT or a Lagrangian Relaxation over multiple time points). However, since the proposed PE fusion tree is batched using all output CTP tracks, a new association and an updated MoP estimation is performed for each PE node. This has the advantage of providing local performance evaluation and avoids the harder problem of determining an association of a CTP track to a truth entity over all time. For example, for crossing entities where the only ambiguity is at the cross, the CTP track may switch truth entities at that point and then, for the full time batch approach, would have, what appears to be many incorrect associations thereafter.

Cumulative PE nodes are added to meet the requirements for the integration of the local evaluations over time, platforms, data types, and scenarios. The MoPs at each time point are integrated in the Cumulative PE Node, as shown, to estimate full scenario MoPs. Performance criteria integration issues will be described later. An alternative tree design is to perform the fused track association based upon an historical PE associations and states (e.g., using confirmed track-to-truth associations) and then to update the current and cumulative MoPs sequentially. This would be shown in the sample PE network Figure by removing the Performance Evaluation node and doing both the current and the cumulative association and MoP state estimation in the “Cumulative PE Node” shown. However, to simplify the PE node processing for our case study, the baseline tree design is to have a separate Cumulative PE node.

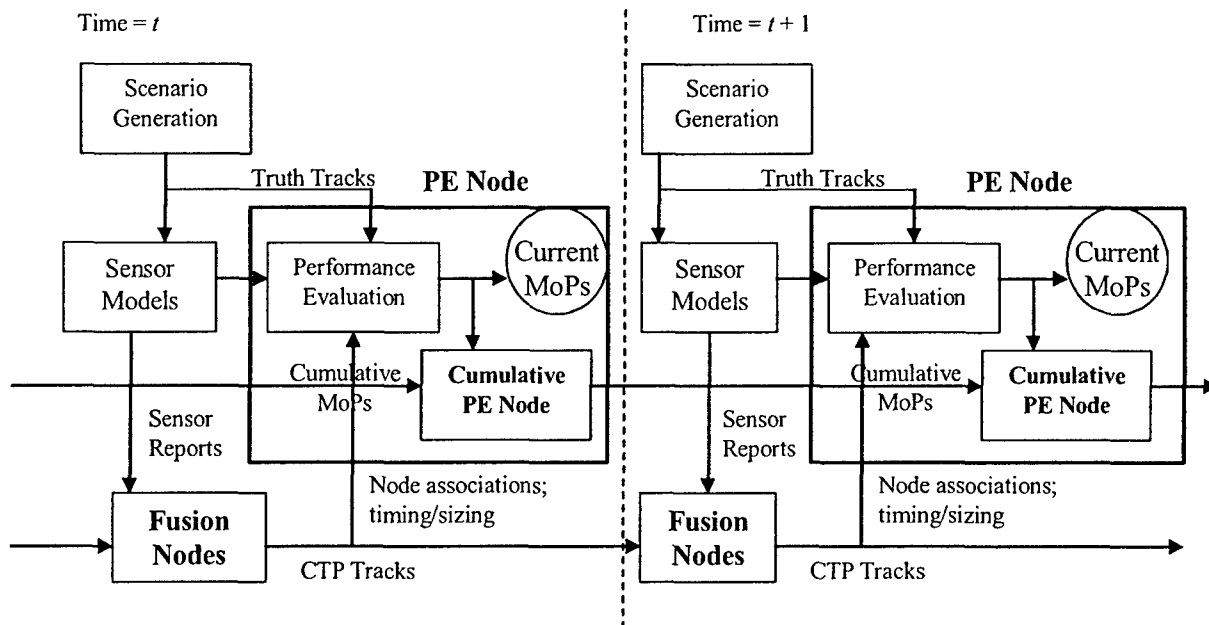


Figure 2-3: The PE Fusion Tree Is Partitioned by Time Enabling Full CTP Evaluation Against Truth.

One of the issues in PE network design is how often to allow track-to-truth associations to switch (e.g., from track A associated with truth 1 to track A associated with truth 2). Switching strategies become important for fusion MoPs where the length of time that a target is in track or tracks on different platforms are consistent impacts the mission. The track-to-truth association process for each track batch shown in Figure 2-3 is independent of the processing of the prior time intervals since the PE nodes do not store and carry ahead any piece of information for its next recursion (or instantiation) as would be possible in the alternative just described above.

Another PE network would be needed to implement a “No Switch Strategy” that requires solving the track-to-truth association problem for the whole scenario all at once. This could be done by optimizing the association hypothesis scores for all feasible multiple time point batches within one fusion node such as depicted in Figure 2-4. Another approach is to apply this “no switch” restriction only over time windows in the scenario. These can be shifting windows or fixed in front of an engagement time. A window of size 5 in a single fusion node is shown in Figure 2-5.

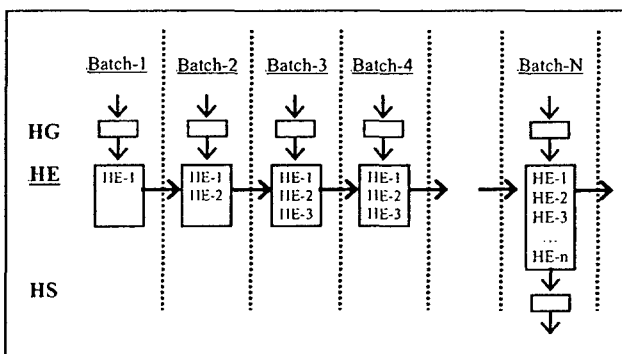


Figure 2-4: No Switch Strategy PE Node.

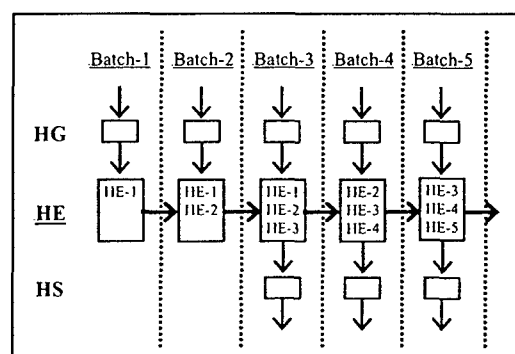


Figure 2-5: Windowed Switch Strategy

The time-batched recursive PE network structure implemented in our initial research is shown in Figure 2-6. In this PE network there was a PE node for the ownership fused track files generated on each of 2 fighters and a PE node for the consistency of the ownership track files at each time point. The data preparation, track association, and MoP state estimation components of each of these PE nodes are shown. The cumulative PE node which had trivial data preparation and association is represented by one PE node box for each of the 3 PE nodes at each time point. Note that different track-to-truth association approaches can be applied in each PE node at each time point (e.g., during ingress, attack, egress) as necessary.

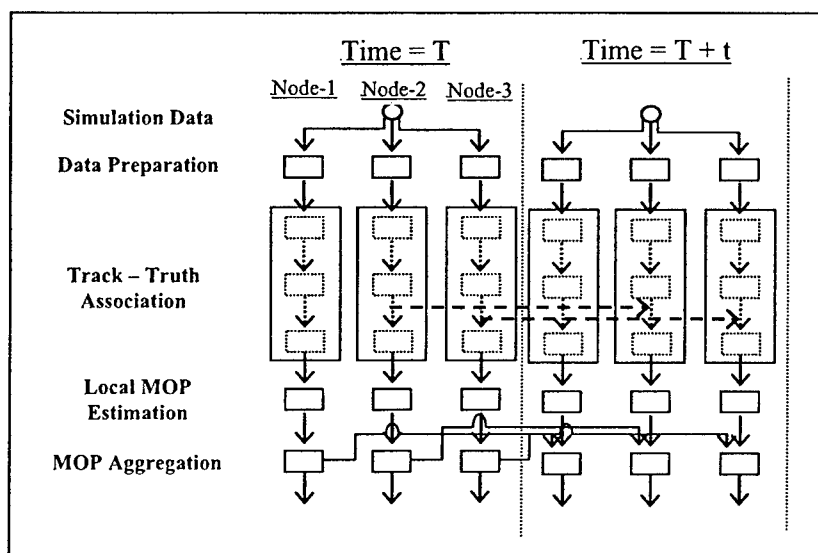


Figure2-6: Case Study PE Node Structure

Figure 2-7 shows a sample PE network (i.e., a tree in this case) for performance evaluation of distributed platform fusion consistency (a notional case of aircraft, ISR platforms, and UAVs is shown). This PE network can be used to perform track file consistency evaluations across platform types. There are numerous PE networks that can be tailored for each distributed fusion SUT, scenario, fusion MoP, etc.

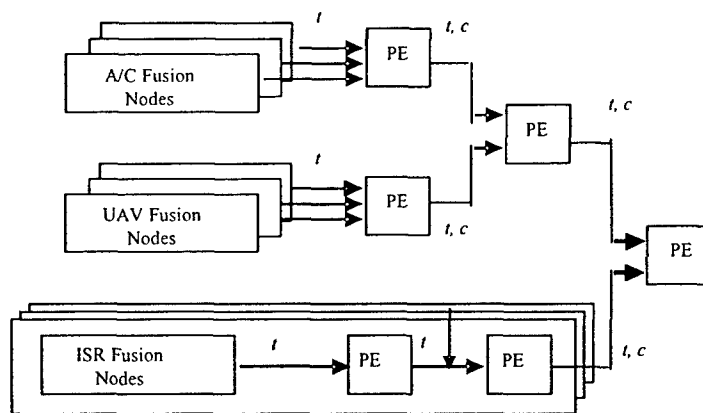


Figure 2-7: Sample PE Network (Tree) for Distributed Platform Fusion Consistency

In summary, PE nodes can be batched across DF&RM SUT levels (e.g., signal, entity, relationship, COA impact), MoPs, scenarios, time, SUT nodes, and entity type (e.g., air, ground, sea, space). PE nodes may also be tailored to the type of DF nodes being assessed (e.g., for sensor, platform, & DF level metrics). PE temporal window size may be mission or weapon dependent (e.g., continuous track fragments for launch). PE nodes perform track-to-truth association to generate accuracy metrics (e.g., coverage, kinematics, ID). PE nodes perform track-to-track association to generate distributed fusion consistency MoPs (e.g., for internetted operations requiring common situation awareness). PE nodes may need probabilistic track-to-truth association for DF test articles with numerous false tracks.

2.2.3 PE Fusion Network Optimization

The PE MoPs are used to perform PE network evaluation to support feedback PE network optimization of the design. This is typically done qualitatively for PE network design and can be done quantitatively for PE node design evaluation to be described in the next Section.

2.3 PE Node Optimization

2.3.1 PE Node Requirements Refinement

In this segment of the PE solution methodology the PE MME/MoE/MoPs from the PE Network Refinement are further refined for each PE node in the network designed above.

2.3.2 PE Node Design

2.3.2.1 PE Node Common Referencing

This segment of the PE node development methodology contains the data mediation, coordinate transformations, misalignment compensations, and time propagations as needed to support PE data association.

2.3.2.2 PE Node Track Association

Hypothesis Generation

In this function the PE node will gate out infeasible track-to-truth associations. Standard techniques are used here.

Hypothesis Evaluation

First a comparison of association hypothesis evaluation scoring schemes extracted from the CMIF Scoring report is given herein. Then a tailoring to PE is described.

The most widely used rigorous scoring approaches are the max a posteriori (MAP) criteria for data association and state estimation. The most common of these is the deterministic association. This standard MAP deterministic data association criterion is used to select the “best” hypothesis that is then used to generate the MAP estimate of the system state. The second scoring approach updates the track state confidence for each report based upon its relative association confidence score. This has been termed probabilistic data association [Ref 15]. The third criterion is the joint optimization over state and association hypotheses. The three MAP scores are defined as follows:

- Deterministic Data Association, then Target Estimation

$$\max_H P(H | \text{reports}) = \max_H [P(\text{reports} | H)P(H)] \text{ then } \max_\theta P(\theta | H, \text{reports})$$

- Target State Estimation with Probabilistic Data Association

$$\max_\theta P(\theta | \text{reports}) = \max_\theta \left[\sum_H P(\theta | \text{reports}, H)P(H | \text{reports}) \right]$$

- Joint Association Decision and Target State Estimation

$$\max_{H, \theta} P(H, \theta | \text{reports}) = \max_H \left[\max_\theta P(\theta | \text{reports}, H) \right] P(H | \text{reports})$$

where H is the association hypothesis and θ is the object state estimate.

The 1st deterministic association strives to decrease the error probability of track estimation by eliminating data outliers, which are data observations that lie outside a specified confidence interval, typically 0.95 or 0.99. Two common techniques used to eliminate outliers are establishing a figure of merit (FOM) and gating. Both of these techniques work by selecting only those data observations that lie within a predetermined error threshold. One way to measure the distance between an established track for a target and a single observation in question is the Mahalanobis distance. This is the measured distance normalized by measurement and track error variances. The Multiple Hypothesis Tracker (MHT) works with deterministic association to handle multiple sensor types, multiple platforms, out-of-sequence reports, and both kinematic and attribute-based sensors. The multiple hypothesis method allows the consideration of lower confidence scenes caused by lower confidence associations. When an association is ambiguous, multiple models are created and a collection of likely hypotheses are selected (i.e., creating what is called multiple scenes). When new data arrives the prior scenes confidence are modified causing pruning of lower confidence scenes, new scenes, and updates of prior scenes.

The 2nd Probabilistic Data Association (PDA) is an approximation to the optimal recursive Bayesian data association strategy. The PDA is a single target algorithm, so each track is filtered in isolation, and it is assumed that any measurements due to other targets can be lumped into the clutter. The PDA enforces the single measurement assignment constraint, namely each target track is only allowed to form at most one measurement. There are two forms of the PDA, known as the parametric and

nonparametric PDA. In the parametric PDA, it is assumed that the rate parameter of the clutter density (λ) is known. In the nonparametric PDA, the rate parameter is unknown and is approximated using $\lambda_{approx} = m/V$. An alternative nonparametric PDA in [14] uses the approximation $\lambda_{approx} = n/A$ where A is the area of the entire surveillance region. This has been extended to a multiple target filter termed the PDAF. The 3rd MAP approach defined above is a hybrid of the above 2.

The deterministic MAP score (i.e., the term "score" is used since it is not necessarily a probability) of the data association hypothesis, H , given the report data, R , is as follows:

$$\max_H P(H|R) = \max_H P(R|H) P(H) / P(R) = \max_H P(R|H) P(H) \quad (1)$$

where,

$P(R|H)$ is the probability density of the reports given H

$P(H)$ is the a priori probability of the association hypothesis, H ,

$P(R)$ is the a priori probability of the reports which is independent of H .

It is convenient to utilize the independent nature of the operations of most platforms. If one assumes independent tracks, $P(R|H)$ can be computed as a product. The score can also be computed recursively over time. This is typically done for time ordered kinematics reports, $Y_j = \{Y_j(0), Y_j(1), \dots, Y_j(T)\}$, as follows:

$$P(Y_j|H) = \prod_{t=0}^T P(Y_j(t)|Y_j(0), \dots, Y_j(t-1), H) \quad (2)$$

At any one point, the overall MAP report-to-track score is the product of three MAP individual scores, which consists of the following three report-to-track score components:

1. Kinematic scoring: $P(Y)$, usually a product of Gaussian density points,
2. Parametric/attribute scoring: $P(Z)$, a sum of class confidences, $P(K)$, times the priors for the attributes,
3. *A priori* hypothesis scoring: $P(H)$ as a product of association hypothesis types.

These three scores can be computed as follows:

$$\max P(H|R) = \max \{P(R|H) P(H)\} = \max \{P(Y|H) P(Z|Y,H) P(H)\} \quad (3)$$

$$= \max \left[\prod_{t=0}^T \{P(Y(S)|Y(T),H) P(Z(S), Z(T)|Y(S), Y(T), H)\} P(H) \right] \quad (4)$$

where

- The maximization's are over all association and non-association hypotheses, H ,
- H is the set of feasible association or non-association hypotheses,
- R are the central track and sensor report data,
- Y is the set of kinematics from both,
- Z is the set of all parameters and attributes from both which are not available,
- The product is over all independent labeled track, T , hypotheses (i.e., of all 5 types),

- $Y(T)$ are the track kinematics, the $P(Y(T)|H)$ term is dropped as constant with respect to the maximization,
- $Y(S)$ are the sensor report kinematics,
- K are the elements of the disjoint class tree,
- $Z(T)$ are the parameters and attributes from the track,
- $Z(S)$ are the parameters and attributes from the sensor report,
- $P(H)$ is the *a priori* confidence in the hypothesis.

These three scores are defined in more detail below.

(i) Kinematics Association Scoring

The association hypothesis kinematic scoring for a new incoming sensor report, $Y(S)$ to an existing track, $Y(T)$ assumes a multivariate Gaussian distribution [ellipsoid], with a central track covariance P which models the error in the track location due to possible motion. Then the kinematics score is computed as follows:

$$P(Y(S)|Y(T), H) = \{1/(2\pi)^{d/2} |V|^{1/2}\} \exp[-1/2 \{I^T V^{-1} I\}] \quad (5)$$

where

- $Y(S)$ are the sensor report Gaussian kinematics with covariance R ,
- $Y(T)$ are the track Gaussian kinematics with covariance P ,
- H is the hypothesis that the report and track are associated,
- d is the dimension of the Gaussian kinematics state,
- $|V|$ is the determinant of the innovations covariance, $V = [\phi P \phi^T + Q] + R$,
- ϕ is state transition matrix, Q is the noise covariance, and the measurement matrix, H , is the identity,
- I is the innovations vector, $I = Y(S) - Y(T)$.

When all the covariances remain constant then the first term can be dropped. This yields the classic Mahalanobis distance measure in the exponent after taking the log and multiplying by (-2). When doing so these conversions also need to be applied to the non-commensurate and *a priori* scores given below.

(ii) Parametric/Attribute Association Scoring

The second term is computed as the product of commensurate attributes and non-commensurate. The commensurate Gaussian parametric data (e.g., both sensors measuring RF, PRI, and/or signature) are computed similar to the kinematic terms above. The non-commensurate attributes (e.g., radar or IR signature and emitter parameters) are independent when conditioned upon the class of the object. Thus, their association compatibility is computed using the probability of the disjoint object classes that they imply. This, non-commensurate score measures the similarity in the platform classifications implied by dissimilar source data. This is accomplished using a disjoint class tree breakdown defined *a priori* according to the ID capability of the sources for each fusion node. To use non-commensurate scoring requires the attributes and parameters, Z , in the report and track data to be independent when conditioned on the feasible platform ID classes. Namely, information about $Z(T)$ does not help estimate $Z(S)$ when the platform class K is known for each class K . Under this assumption for each report and track pair, the second term scores an object track ID tree with a sensor report ID tree as follows:

$$P(Z(S), Z(T)|Y(S), Y(T), H) = P(Z(S)|Y(S), H) P(Z(T)|Y(T), H) [\sum_K \{P(K|Z(T), Y(T), H) \times P(K|Z(S), Y(S), H) / P(K|Y(T), Y(S), H)\}] \quad (6)$$

where:

- The first two terms in front of the sum are constant with respect to the maximization when they appear in every hypothesis (i.e., as they do in this option, so they are ignored here), also the kinematics conditioning has been restricted to each report and track, respectively,
- The first term after the sum is the class K element of the object track ID disjoint class tree, since the conditioning on $Y(S)$ can usually be dropped due to $Y(T)$,
- The second term after the sum is the class K element of the sensor report ID tree corresponding to that object since the conditioning on $Y(T)$ can usually be dropped due to $Y(S)$,
- The third term after the sum (i.e., in the denominator) is the *a priori* probability of that class K [Note: when denominator is 0 for an ID class K , then whole term in the equation sum is 0], and
- The term components are as described above.

The class tree for each sensor and each report is conditioned on only its own kinematics and attributes. Thus, it is derivable from each sensor individually. Also, when either the report or track noncommensurate attributes do not contribute to the ID, these non-commensurate terms in the equation sum to one (i.e., the class K terms in the tree are disjoint and cover all possibilities). This term only rigorously applies when the current sensor report attributes are non-commensurate with the track attributes. If previous report attributes have already been fused (i.e., integrated) with the track attributes, then these previous attributes would implicate corresponding attributes in the current report even given the platform class K . Thus the report and the track attributes would be commensurate. When such attributes are available, it is better to use the commensurate scoring in both the report and track (e.g., pulse descriptors, IR signatures, etc.).

(iii) *A Priori* Association Hypothesis Scoring

An association hypothesis is composed of the following types of hypotheses: (1) Association, (2) Report on pop-up object (i.e., a track initiation), (3) False Alarms, (4) Track Propagation, and (5) Track Deletions. $P(H)$ is the probability of H computed using the following (as available):

- Probability of detection and false alarm statistics
- Number of reports from each source
- Source field-of-view, operating mode, and conditions
- *A priori* scene descriptors and probability of redetection
- Object birth and death statistics

The *a priori* hypotheses terms, $P(H)$, use the following approximate scoring equations for each sensor report S and track T hypothesis :

$$\begin{aligned} P(\text{association}) &= [1 - P_{FA}(S)][1 - P_{FA}(T)]P_D(S)P_D(T) \\ P(\text{pop-up}) &= [(1 - P_{FA}(S))(1 - P_D(T))P_D(S)] \\ P(FA) &= [P_{FA}(S)P_D(S)] \\ P(\text{propagate}) &= [(1 - P_{FA}(T))(1 - P_D(S))P_D(T)] \\ P(\text{drop}) &= [P_{FA}(T)P_D(T)] \end{aligned} \quad (7)$$

where

- $P_D(S)$ is the probability of detection of this object reported by the sensor, which is determined by sensor testing. Its primary use is in scaling the probability of track propagation, since it appears in all of the report hypotheses. It is estimated as the probability of redetection for the association hypothesis, and as a result is usually high (e.g., $>.9$). In the hypothesized case of an initial detection by a sensor, the term in the pop-up, FA, and propagate hypotheses is the probability of detection of a new object.
- $P_{FA}(S)$ is the probability of false alarm (FA) of the sensor for this type of report, which is also determined by sensor testing. It can be approximated as the expected number of false alarms (i.e., under these report conditions) divided by the number of detected objects plus this expected number of false alarms over the field of view (FOV),
- $P_D(T)$ is the probability of detection of this object in the central track file, which is the combined probability of detection of this object by any of the sensors contributing to the track file (i.e., as updated in the last fusion node using the equations in the state update) multiplied by $[1 - P(\text{new object appearing during this time interval})]$. If the former term is very near one, then this term is dominated by the $[1 - P(\text{new object appearing during this time interval})]$ term. Poisson arrival statistics, if available, are used here. If the probability of false alarm is low enough that a detection starts a track, then the value for $P_D(T)$ from the last fusion node can be used as defined in the state update.
- $P_{FA}(T)$ is the probability that this track is a false alarm, which can be estimated by maintaining the track existence confidences over time plus considering the probability of track death during this time interval. The former FA probability will usually decrease over time due to increased tracking confidences. If this resulting track confidence is very near one, then this term is dominated by the probability of track death (i.e., dying in the field of view (FOV) or moving out of the FOV). This is where Poisson track death statistics, if available, are used. The updated value for this term from the last fusion node is used.

For the non-association report hypotheses (i.e., pop-up initiation, and false alarm) the expected value of the kinematics score is used. Namely, the kinematics score equation is used except that the chi-square statistic (i.e., $\{I^T V^{-1} I\}$) is replaced with its mean μ . Namely,

- $\mu = .455$ for 1 degree of freedom (DOF) (e.g., bearings-only)
- $\mu = 1.39$ for 2 DOF (e.g., x and y)
- $\mu = 2.37$ for 3 DOF (e.g., Cartesian (x, y, z))
- $\mu = 3.36$ for 4 DOF (e.g., 2 dimensions with rates)
- $\mu = 4.35$ for 5 DOF
- $\mu = 5.35$ for 6 DOF (e.g., Cartesian (x, y, z) with rates)

Also, for the non-association report hypotheses the innovations covariance is the report covariance, R , for which the inverse square root of the determinant is taken for the up-front multiplier in the kinematics scoring equation. The noncommensurate term for the non-association report hypotheses is constant with respect to the maximization, since the class tree term sums to one. So it can be ignored. Thus, the non-association report hypothesis score is the product of their (i.e., pop-up and false alarm) a priori score given above and their kinematics term with the above two values used in its " V " innovations terms.

For the non-association track hypotheses (i.e., propagation, and drop track), the kinematics, $P(Y(T))$, and noncommensurate terms are all constant with respect to the maximization. Thus, all the non-association hypothesis scores have just the above a priori terms, except the pop-up and the report false

alarm which all have only the additional expected value kinematics multiplicative term. Each association hypothesis has all the three of the terms defined above, where the non-commensurate term is unity whenever either the report or the track do not provide a platform ID tree.

Max *A Posteriori* Hypothesis Optimization Summary

The total scene hypothesis score is the product of the individual hypothesis scores for how all the given batch of reports and tracks are associated (i.e., for each of the 5 types of hypotheses). An example is as follows:

- Association Hypotheses

$$P(Y(S)|Y(T),H) P(Z(S), Z(T)|Y(S), Y(T), H) P(H) = \{|V|^{-1/2}\} \exp[-1/2\{I^T V^{-1} I\}] \times \{\sum_k [P(K|Z(T),Y(T), H) P(K|Z(S),Y(S), H)/P(K|Y(T),Y(S), H)]\} \times [1-P_{FA}(S)] [1-P_{FA}(T)] P_D(S) P_D(T) \quad (8)$$

2. Pop-up (i.e., Track Initiation) Hypotheses

$$P(Y(S)|Y(T),H) P(Z(S), Z(T)|Y(S), Y(T), H) P(H) = \{E(|V|^{-1/2})\} \exp[-1/2\{\mu\}] \times [1-P_{FA}(S)] [1-P_D(T)] P_D(S) \quad (9)$$

3. False Alarm (FA) Hypotheses

$$P(Y(S)|Y(T),H) P(Z(S), Z(T)|Y(S), Y(T), H) P(H) = \{E(|V|^{-1/2})\} \exp[-1/2\{\mu\}] \times P_{FA}(S) P_D(S) \quad (10)$$

4. Propagation Hypotheses

$$P(Y(S)|Y(T),H) P(Z(S), Z(T)|Y(S), Y(T), H) P(H) = [1-P_{FA}(T)] [1-P_D(S)] P_D(T) \quad (11)$$

5. Track Drop Hypotheses

$$P(Y(S)|Y(T),H) P(Z(S), Z(T)|Y(S), Y(T), H) P(H) = P_{FA}(T) P_D(T) \quad (12)$$

In PE these a priori probability of detection and false alarm values are summarized as follows:

- $P_D(S)$ is the probability of detection of a truth track in the CTP track file (i.e., probability that a truth will appear in the CTP track file). To begin with we will specify a finite constant probability of the fusion output CTP track file not containing a truth entity that should have been detected (e.g., the CTP Pd value). This term will then be multiplied by the $[1 - P(\text{a truth should not appear in the CTP track file (e.g., since it is new during the last time interval or not detectable yet)})]$. New truth arrival statistics and sensor coverage statistics are used here. When available the values of this term estimated by the fusion system will be used.
- $P_{FA}(S)$ is the probability that this CTP track is a false alarm, which to begin with is estimated to be the CTP Pfa for track S . This value is then multiplied by the probability of truth track death (i.e., dying in the field of view (FOV) or moving out of the FOV) before the CTP drops track. Track death statistics and sensor false alarm statistics are used here. Also, when available the values of this term estimated by the fusion system will be used.
- $P_D(T)$ is the probability that a valid CTP track will appear in the truth track file. This is presumed to be 1.0 to begin with (i.e., the truth file contains all valid entities).
- $P_{FA}(T)$ is the probability that truth is a false alarm (FA). To begin with this equals 0.0.

The CTP false hypothesis score is the product of its a priori score given above and its kinematics term with the above value for μ . For the non-association truth propagation hypothesis (i.e., fourth line of

equation), the kinematics, $P(Y(T))$, and noncommensurate terms are all constant with respect to the maximization. Thus, all the non-association hypothesis scores have just the above a priori terms, except the CTP false hypotheses that have the additional expected value kinematics multiplicative term. A key research issue here is what to use for this expected value when it is not constant. For PE, the $E(|V|^{-1/2})$ can be approximated by using the weighted average of all the feasible track to truth association V 's where the weights are the values in the association hypothesis matrix along the track row.

In summary, the total scene hypothesis score to begin with is the product of the individual hypothesis scores for how all the given batch of CTP tracks and truth tracks are associated (i.e., for each of the 5 types of hypotheses). In effort the typing is not significant to begin with, so the equations become as follows:

➤ Association Hypotheses

$$P(Y(T)|Y(S),H) P(H) = \{|V|^{-1/2}\} \exp[-1/2\{I^T V^{-1} I\}] \times [1-P_{FA}(S)] [1-P_{FA}(T)] \times P_D(S) P_D(T) \quad (13)$$

➤ CTP Pop-up of Non-Truth Hypotheses (for spiral 1, probability = 0) (14)

➤ CTP False Hypotheses

$$P(Y(T)|Y(S),H) P(H) = E(|V|^{-1/2}) \exp[-1/2\{\mu\}] \times P_{FA}(S) P_D(S) \quad (15)$$

➤ Truth Propagation Hypotheses

$$P(Y(T)|Y(S),H) P(H) = [1-P_{FA}(T)] [1-P_D(S)] P_D(T) \quad (16)$$

➤ Truth False Alarm Hypotheses (for spiral 1, probability = 0) (17)

Hypothesis Selection

Many fusion problems are solved by first selecting the “desired” association of the data and using this association to update the state. This deterministic data association then becomes a nonlinear labeled set covering problem (a subclass of 0-1 integer programming problem).. This deterministic data association problem formulation using max a posteriori scoring is defined as follows:

Let $J = \{j | \lambda^j \text{ is a labeled feasible track}\}$ require the hypothesis H to satisfy $H \subseteq J$ and $\bigcup \lambda^j = \{\text{reports}\}$

Assume that for any H in the $\{\lambda^j | \text{reports}\} \forall j \in H$ are independent, then the $\max_H P(H | \text{reports})$ is the solution to the linear set covering problem.

$$\begin{aligned} & \min_x \sum_{j \in J} P_j X_j \\ \text{where } & \sum_{j \in J} A_{ij} X_j \geq 1 \\ & P_j = -\log[P(\lambda^j | \text{reports})] \geq 0 \\ & X_j = \begin{cases} 1 & \text{if } j \in H \\ 0 & \text{otherwise} \end{cases} \quad \forall j \in \{\text{all current feasible tracks}\} \\ & A_{ij} = \begin{cases} 1 & \text{if } i \in \lambda^j \\ 0 & \text{otherwise} \end{cases} \quad \forall i \in \{\text{all prior feasible tracks and current reports}\} \end{aligned}$$

if $\lambda^j, \forall j \in H$, is required to be disjoint then an equality constraint is used.

Basically, the hypothesis selection algorithm in data association generates collections of feasible tracks, called candidate scenes, and then selects the scene(s) to be retained and used in state estimation to generate the situation assessment estimate for the user. Set covering search algorithms are selected based upon problem constraints, complexity and performance requirements. Irrevocable decision heuristics are the simplest but worst performing. Breadth-first approaches require more memory; however, they have less computational burden for a given performance level as compared to the depth-first approaches. Many decades of operations research provides a rich heritage of efficient algorithms from which to select from. All of these depend upon robust and accurate HE scoring described above which are much less mature.

Deterministic data association is a standard approach for PE. For this approach, the scene with the "best" (i.e., highest) association score (i.e., product of each of its hypotheses scores), as found in HS, is selected for use in MoP state estimation. The baseline design is a 2D assignment algorithm (e.g., JVC, Munkres, etc.) that will be applied to the association matrix of scores computed as described above. To meet the square matrix requirement for the strict assignment problem solution extra rows or columns will be added to the association score matrix. To do so an extra 'missing CTP track' row will be added for every truth track over the number of CTP tracks. Scores in these rows will be the CTP false hypothesis scores above. For the case of more CTP tracks than truth tracks, an extra 'false alarm truth' column will be added for every extra CTP track. The scores in these columns will be the truth propagation hypothesis scores above.

2.3.2.3 MoP State Estimation per Fusion Node

The fusion node refined MoPs are used to ascertain how well the distributed fusion subsystem achieves its performance goals and as such form the basis for the each PE node state estimation. Typical MoPs are as follows:

1. Average CTP track position accuracy
2. Percentage of conflicting CTP track types
3. CTP sensor coverage preservation
4. CTP coverage improvement ratio over any one sensor/source
5. Average CTP probability of false alarm
6. Percentage of correct fusion system associations
7. Average time from detection receipt to CTP update

A traditional CTP kinematics accuracy MoP is the standard deviation over time of the error in the CTP updates after detection averaged over all tracks for each test case. This traditional measure is the second moment of the associated CTP location state error density. It is based upon the first moment (i.e., mean error). The mean error in the track estimate, preferably in independent coordinates for each warfighter CTP platform and for each associated track entity, T_i , at each selected time point (e.g., CTP update times) in each scenario is computed as the average of the individual errors:

$$\text{Platform } j \text{ error in entity } T_i \text{ (time } t) = \{\text{true location of the entity associated with } T_i\}(\text{time } t) - \text{platform } j \text{ CTP tracker location estimate for } T_i \text{ (time } t) \quad (18)$$

These errors can be computed at each CTP PE evaluation time for each track, T_i . This may require propagation of the truth track. For evaluation where the CTP track propagation accuracy is also important the difference above is taken at the current time. For tracks where the current time is not the

CTP update time, this requires that the CTP updated state be subtracted from the associated truth track at that time. In both cases for detailed PE, it is useful to plot the mean error over time for each dimension (e.g., in position and velocity) with the standard deviation of the track error as computed in the CTP tracker for each track output from each platform in each scenario. Even without Monte Carlo runs, the number of kinematics error and standard deviation plots over time is very large (e.g., $[# P, V \text{ dimensions}] \times [\# \text{ tracks}] \times [\# \text{ fighter platforms}] \times [\# \text{ scenarios}]$). Thus, it becomes desirable to utilize an overall performance measure to capture large portions of the kinematics error information.

Error combining approaches include the standard deviation, the weighted RSS error measure, and the weighted average performance (i.e., $\{\sum_k (|\text{error}(k)| / \sigma(k))\} / K$). The off-diagonal covariance terms (i.e., joint moments) can also be estimated, if needed (e.g., if highly range dependent errors off-axis exist). The standard deviation (the second moment) is then the square root of the following:

$$\text{Variance of the error} = \{\sum_j (\text{measured error}(j) - \text{mean error}(j))^2\} / J \quad (19)$$

where the sum is over updated CTP tracks and over time for each test case. If the CTP track covariance, P , is trusted, then the additional measure below provides the covariance weighted RSS error:

$$\text{Weighted RSS Performance} = \{\sum_j (\text{error}(j)^2 / P(t)(j))\}^{1/2} \quad (20)$$

where the sum is over the true entities, j , and $P(t)$ is the updated covariance of the CTP track states. This latter measure is used when sensors have different errors that need to be weighted accordingly (e.g., errors are of equal importance).

In all the above approaches the weighting of the error parameters (e.g., over dimensions, time, entities, platforms, or scenarios) is *ad hoc* (e.g., uniform or by standard deviation). Similar problems occur with integrating the error standard deviations. This leads to replacing the standard deviation weightings on the errors with a constant required sigma for each k and as necessary specifying the requirements on the error standard deviations. In fact once requirements are brought into the performance measure, a generally recommended approach is to identify tracking accuracy requirements for each entity, time, etc. For example one can establish a weapons requirements basket or surveillance window that the track estimate is required to be within (e.g., being 10 times more accurate than a given window is not 10 times more valuable). In this case the performance can be computed based upon one (or a few) runs by integrating the CTP output track density (i.e., estimated mean and covariance) over this window (e.g., centered at the true entity location) to yield the probability of satisfying the user track accuracy requirements. These probabilities can then be combined rigorously over all entities, time, platforms, and (if desired) scenarios. If Monte Carlo testing is used (e.g., thousands of runs per scenario) then the average of the above probability of satisfying user requirements can be computed and compared to the count over the Monte Carlo runs of the percentage of times that the requirements are met. However, this approach requires that the kinematics accuracy requirements be specified. What is needed in many cases is a summary PE measure that can be recursively updated on-line during the distributed fusion system evaluation.

2.3.3 PE Node Optimization

This segment of the PE node development methodology applies the refined measures to assess the current point design. The results of this assessment are feedback for improved PE node design or possibly improved needs refinement.

3. Mapping Guidelines of PE Solutions to Problem Space

Problem-to-solution mappings provide a segmentation of the problem and solution spaces and rules of thumb for which areas of problem space are most applicable to each area of solution space. As such, the problem-to-solution space mappings help the PE designer to decide which PE solution is most useful for each PE problem. For example, these mappings help the user decide when to batch PE inputs over time and SUTs; or when to use deterministic, MHT, and probabilistic, track-to-truth and track-to-track association. In the following subsections we provide tabular representations of problem-to-solution space mappings for the Hypothesis Generation, Hypothesis Evaluation and Hypothesis Selection subfunctions of the central Data Association function in a PE node portion of the PE framework solution space described in Section 2. Further work is recommended to extend these problem-to-solution mappings and possibly provide an automated tool for more affordable PE system design especially for novice PE designers.

Also further work is needed to determine problem-to-solution mappings for the PE network and the other areas of PE node solution space defined in Section 2. For example, the joint probabilistic data association (JPDA) approach, is best applied where the associations are usually not clear such as when there is significant clutter and false tracks. In a situation where a platform's radar is generating a lot of radar clutter (e.g., in an urban environment or with significant countermeasures), JPDA can be used to estimate the tracking error to the truth. A deterministic association might not give a fair representation of the error since it may choose false tracks to associated to the truth entities. Another strategy would be to use multiple scans and use the Lagrangian relaxation approach. Extensions to N-D Lagrangian Relaxation for many-to-one associations can be used to handle the low resolution radar cases versus truth entity separations. Track-to-truth bias misalignment estimation based upon confirmed track-to-truth associations can be applied to provide better track accuracy MoPs. The track ID pedigree can be used to remove significant error correlations and thus improve track-to-track associations for consistency MoPs.

3.1 Association Hypothesis Generation Problem-to-Solution Space Mappings

A sample problem-to-solution mapping for hypothesis generation is shown in Figure 3-1.

| SOLUTION SPACE | Single Scan | Multiple Scans | Batch | Single Hypothesis | Multiple Hypotheses | Unique Assignment | Non-Unique Assignment | Geographic Prior | Generalized Dist | Mahalanobis Dist | Probabilistic | Ad hoc | Location Only | Kinematic | Target Attributes | Target Identity |
|------------------------------|--------------|----------------|-------|-------------------|---------------------|-----------------------|-----------------------|-------------------|------------------|------------------|---------------|--------|-----------------|-----------|-------------------|-----------------|
| PROBLEM SPACE | No. of Scans | | | Hypotheses | | Assignment Uniqueness | | Evaluation Metric | | | | | Gating Strategy | | | |
| NATURE OF INPUT DATA | | | | | | | | | | | | | | | | |
| • Location Data | | Y | Y | Y | Y | | Y | Y | Y | Y | Y | Y | Y | Y | | |
| • Attribute Parameters | | Y | Y | Y | Y | | Y | N | Y | Y | Y | Y | | | Y | |
| • Target Class/ID | | Y | Y | Y | Y | | Y | N | Y | N | | Y | | | Y | Y |
| • High Observation Rate | | Y | Y | Y | | | Y | | | | | | | | | |
| TARGET CHARACTERISTICS | | | | | | | | | | | | | | | | |
| • Location Predictability | | Y | | Y | Y | | Y | N | | Y | Y | | | Y | | |
| • Identity Attributes | | Y | Y | | | | Y | | | Y | Y | Y | | | Y | Y |
| • Maneuverability | N | Y | Y | N | Y | | Y | | | | | | | | | |
| TARGET DENSITY | | | | | | | | | | | | | | | | |
| • High Density | N | Y | Y | N | Y | | Y | | | | | Y | | | | |
| • Low Density | v | Y | Y | Y | N | | N | | | | | | | | | |
| SENSOR CHARACTER | | | | | | | | | | | | | | | | |
| • High PFA | N | Y | Y | N | Y | | Y | | | Y | Y | | | | | |
| • Known Obs. Statistics | | Y | Y | Y | Y | | Y | | | Y | Y | | | | | |
| • Complex Signal Propagation | N | Y | Y | N | Y | Y | Y | | | | | | | | | |
| • Low Spatial Resolution | | Y | Y | N | Y | N | Y | | | | | Y | Y | | | |
| AVAIL PROCESSING/DECISION | | | | | | | | | | | | | | | | |
| • High Processing/Long Time | | Y | Y | Y | Y | Y | Y | | | Y | Y | | | | | |
| • Low Processing/Short Time | Y | N | N | N | N | Y | Y | v | Y | | N | Y | | | | |

Figure 3-1: Hypothesis Generation Problem-to-Solution Space Mapping

3.2 Association Hypothesis Evaluation Problem-to-Solution Space Mappings

The performance versus cost/complexity trade for data association has yielded sundry solutions including the following:

- Simple high confidence only association (e.g., score gating)
- Deterministic association using assignment or set covering algorithms to search for the “best” association confidence scores
- Probabilistic association which updates the track state confidence for each report based upon its relative association confidence score

All these techniques require a methodology (i.e., from simple heuristics to rigorous probabilities) for considering alternative track associations. Popular probabilistic scoring schemes include max likelihood (ML), max a posteriori (MAP), Neyman-Pearson, generalized max likelihood, and chi-square tests, see Figure 3-2.

One major probabilistic scoring trade is between MAP and chi-square scoring. The former is a point on a Gaussian PDF whereas the later just uses the chi-square-square distributed exponent of the Gaussian. The payoff for chi-square is its ability to compare data with differing dimensions. Whereas MAP requires *ad hoc a priori* assumptions on the probability of receiving the extra dimension (e.g., range) data given the association hypothesis, $P(R|H)$. However, chi-square scoring does not enable rigorous

comparisons with non-Gaussian data such as non-commensurate attributes and a priori data. Also the chi-square tends to give too much weight to lower confidence far away tracks as shown in Figure 3-3. The MAP scoring scheme provides for rigorous comparison of multi-spectral data.

Max *A Posteriori* (MAP) and Min Probability of Error:

$$\max_{H_i} P(R|H_i)P(H_i) \sim \min_i P(d_1|H_2)P(H_2) + P(d_1|H_1)P(H_1)$$

Max Likelihood (ML):

$$\max P(R|H_i) \sim \frac{P(R|H_1)}{P(R|H_2)} \begin{matrix} d_1 \\ > \\ < \\ d_2 \end{matrix} \quad \begin{matrix} R = \text{Sensor Reports } X \text{ and } Y \\ H_1 = \text{Association Hypothesis} \\ H_2 = X \text{ and } Y \text{ Not Associated} \end{matrix}$$

Neyman-Pearson (NP):

$$\text{Fix } P(d_2|H_1) = \alpha \text{ Then } \max P(d_2|H_2) \sim \min P(d_2|H_1)$$

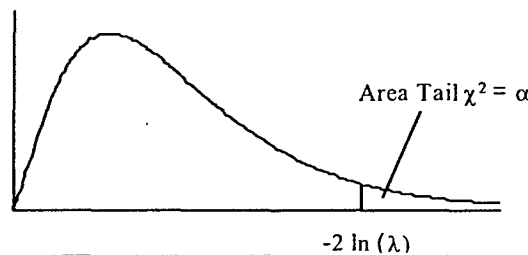
(NP is a Uniformly Most Powerful (UMP) Test α is a level-of-significance
 $\beta = 1 - P(d_1|H_2)$, the power-of-test, is maximized.)

$$\sim \frac{P(R|H_1)}{P(R|H_2)} \begin{matrix} d_1 \\ > \\ < \\ d_2 \end{matrix} \lambda \text{ where } \alpha = \int_{D_2} P(R|H_1) dR \text{ and } D_2 = \{R | P(R|H_1) - \lambda P(R|H_2) < 0\}$$

Generalized ML: (Not Necessarily a UMP Test)

$$\text{Fix } P(d_2|H_1) = \alpha \text{ Then } \frac{\max P(R|H_1)}{\max P(R|H_2)} \begin{matrix} d_1 \\ > \\ < \\ d_2 \end{matrix} \lambda$$

$$\sim (X-Y)^T V^{-1} (X-Y) \begin{matrix} d_1 \\ > \\ < \\ d_2 \end{matrix} -2 \ln(\lambda) \quad (\text{For Gaussian Errors) where } \lambda \text{ is defined by}$$



Chi-Square Tail Test:

$$\text{Fix } P(d_2|H_1) = \alpha \text{ Then Test for Rejection of } H_1: \text{Mean of } (X-Y) = 0$$

$$\sim \int_c^\infty X_n^2(s) ds \begin{matrix} d_1 \\ > \\ < \\ d_2 \end{matrix} \alpha \text{ where } c = (X-Y)^T V^{-1} (X-Y)$$

Figure 3-2: Comparison of Data Association Decision Criteria

• Chi-square (Mahalanobis) Scoring:

- > $|V^{-1}| = [R_1 - T_2]^2 / [\sigma_R^2 + \sigma_{T_2}^2] = 2^2 / [1 + 1] = 2$
- > $|V^{-1}| = [R_1 - T_3]^2 / [\sigma_R^2 + \sigma_{T_3}^2] = 4^2 / [1 + 16] = 16/17$
- > R associated to further away less accurate T_3

• Max. a Posteriori (Bayesian):

- > $[2\pi V]^{-.5} e^{(-.5|V^{-1}|)} = [6.28 \cdot 2]^{-.5}$
 $e^{(-.5[R_1 - T_2]^2 / [\sigma_R^2 + \sigma_{T_2}^2])} \sim .28 e^{-1} \sim .10$
- > $[2\pi V]^{-.5} e^{(-.5|V^{-1}|)} = [6.28 \cdot 17]^{-.5} e^{(-.5[R_1 - T_3]^2 / [\sigma_R^2 + \sigma_{T_3}^2])} \sim .097 e^{-.47} \sim .060$
- > R is associated to the closer more accurate T_2

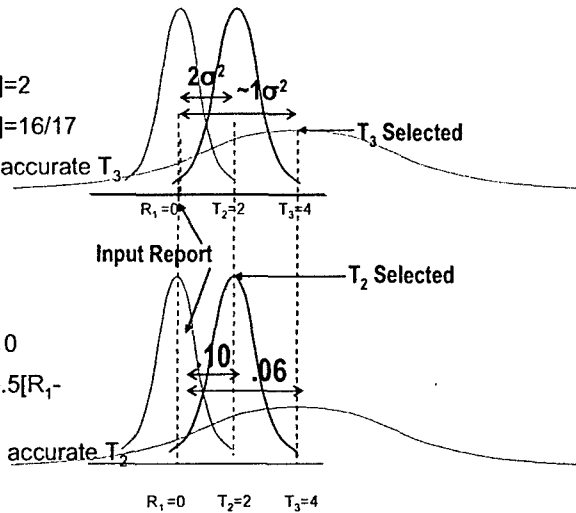


Figure 3-3: MAP Scoring Provides a Better Balance of Nearby High Confidence Data Versus Less Accurate Further Away Tracks

Differences in association scores as report-to-track separation increases for MAP, chi-square (Mahalanobis), and the chi-square integral of the tail approaches in Figure 3-4 shows the penalty for using CHI related versus MAP scores. MAP is especially favored whenever ID and a priori data need to be considered rigorously in the association scores.

| Report-Track Error | Gaussian MAP | χ^2 Integral of Tail | CHI (Mahalanobis) |
|--------------------|--------------|---------------------------|-------------------|
| 0 σ | 1.0 | 1.0 | 0 |
| .1 σ | .995 | .92 | .01 |
| .32 σ | .95 | .75 | .1 |
| .4 σ | .92 | .7 | .16 |
| .675 σ | .796 | .5 | .455 |
| 1 σ | .6 | .32 | 1. |
| 1.15 σ | .5 | .25 | 1.32 |
| 1.6 σ | .275 | .12 | 2.6 |
| 2 σ | .13 | .04 | 4. |

Figure 3-4: Comparison of Alternative Gaussian-Based Association Scoring Techniques

The following provides the conditions where the alternative association hypothesis evaluation techniques should be applied.

- Probabilistic: Preferred if statistics known
 - > Chi-Square Distance
 - Doesn't require prior densities
 - Useful for comparing multi-dimensional Gaussian data
 - However, no natural way to incorporate attribute and *a priori* data

- > Likelihood
 - Doesn't require unconditional prior densities, $p(x)$
 - Does require conditional priors, $p(Z|x)$
- > Bayesian Maximum a Posteriori (MAP)
 - Naturally combines kinematics, attribute, and a priori data
 - Provides natural track association confidence measure
 - However, requires prior probability (e.g. kinematics and class) densities; difficult to specify
- Non-Probabilistic: Useful if high uncertainty in the uncertainty
 - > Evidential (Dempster-Shafer)
 - Non-statistical: User specifies evidence "mass" values (support and plausibility numbers)
 - Essentially 2-point calculus (uniform uncertainty-in-the-uncertainty with simple knowledge combination rules)
 - > Fuzzy Sets
 - User specifies membership functions to represent the uncertainty-in-the-uncertainty
 - User specifies fuzzy knowledge combination rules (e.g., sum, prod, max/min) which are much easier compute than second-order Bayesian
 - More complex to develop, maintain, and extend
 - > Confidence Factors and Other *ad hoc* Methods
 - Explicit derivation of logical relationships
 - Generally *ad hoc* weightings to relate significance of factors
 - Can include information theoretic and utility weightings

Figure 3-5 shows a sample problem-to-solution mapping for hypothesis evaluation.

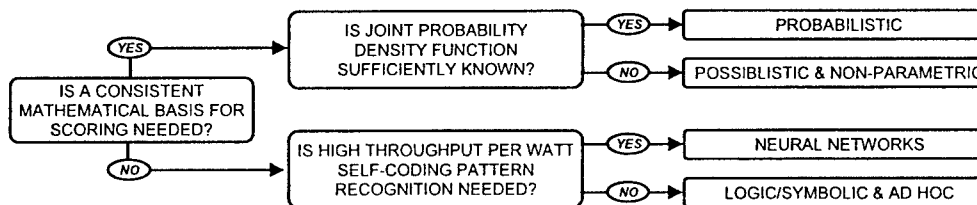
| SOLUTION SPACE | | Probabilistic | | | | | | Possibilistic | | Logic/Symbolic | | | | Neural | | | Unified |
|-----------------------|-----|---------------|-----|----|-----|-----|-----|---------------|-----|----------------|----|----|-----|--------|----|-----|---------|
| PROBLEM SPACE | AdH | Lkl | Bay | NP | Chi | CEA | Inf | DS | Fuz | S/F | SD | ES | C-B | US | FF | Rec | RS |
| INPUT DATA | | | | | | | | | | | | | | | | | |
| • Identity/attributes | | Y | Y | | N | | | | | | | | | | | | Y |
| • Kinematics | | Y | Y | | Y | | | | | | | | | | | | Y |
| • Parameter | | | Y | Y | Y | | | | | | | | | | | | Y |
| • A priori sensor | | N | Y | | N | | | | | | | | | | | | Y |
| • Linguistic data | Y | | | | | Y | | Y | Y | | Y | | | | | | Y |
| • Spatio-temporal | | | | | | | | | | Y | | | | | | | Y |
| • High uncertainty | | | | | | | | Y | Y | | | | | | | | Y |
| • Unknown structure | | | | | | | | | | | | | | Y | Y | Y | |
| • Non-parametric | Y | | | Y | | | Y | | Y | | | | | | | | |
| • Partial data | | | | | | | | | | | Y | | | | | | |
| • Differing | | | | | Y | | | | | | | | | | | | |
| • Differing | | | | | | Y | | | | | | | | | | | |
| • Error PDF known | | N | Y | Y | N | | N | N | N | N | N | N | N | N | N | N | N |
| SCORE OUTPUT | | | | | | | | | | | | | | | | | |
| • Yes/no, pass- | Y | | | | | | | | | | | | | | | | |
| • Discrete score | Y | | | | | | | | | Y | Y | Y | Y | Y | Y | Y | |
| • Numerical scores | | Y | Y | Y | Y | Y | Y | | | | | | | | | | Y |
| • Multi scores per | | | | | | | | Y | | | | | | | | | Y |
| • Confidence | | | | | | | | | Y | | | | | | | | Y |
| PERFORM MEAS | | | | | | | | | | | | | | | | | |
| • Low | Y | Y | Y | Y | Y | N | N | N | N | N | N | N | Y | Y | Y | Y | N |
| • Compute | | | | | | | | | | N | | N | Y | Y | Y | Y | |
| • Score accuracy | N | Y | Y | N | Y | Y | | Y | Y | N | N | N | N | N | N | N | Y |
| • User adaptability | | | | | | | | | | Y | | Y | Y | Y | Y | Y | |
| • Training set | | | | | | | | | | | | | N | N | Y | Y | |
| • Self-coded/trained | | Y | Y | Y | Y | Y | Y | Y | Y | Y | Y | Y | Y | Y | Y | Y | |
| • Robustness to | | | | | | | | | | | | | Y | Y | Y | Y | |
| • Result explanation | N | | | | | | | | | Y | Y | Y | Y | Y | Y | Y | |
| • High processing | N | | N | N | | y | Y | | N | Y | | Y | Y | Y | Y | Y | |

KEY
AdH Ad Hoc
Lkl Likelihood
Bay Bayesian
NP Non-parametric
Chi Chi-Squared
CEA Conditioned
Event Algebra
Inf Information
Theoretic Dempster-Shafer
DS
Fuz Fuzzy Logic
S/F Scripts/Frames
SD Semantic Distance
ES Expert Systems
C-B Case-Based Reasoning
US Unsupervised Learning
FF Feed-Forward
Rec Recurrent Supervised Learning
RS Random Set

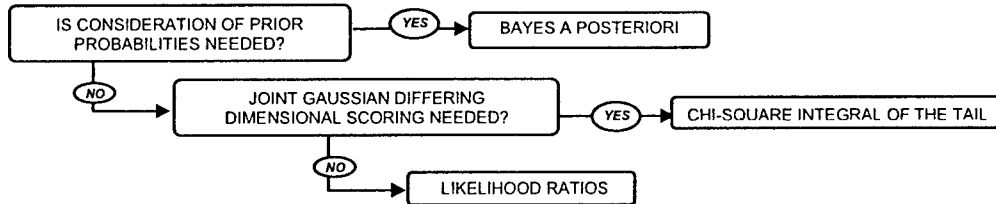
Figure 3-5:Hypothesis Evaluation Problem to Solution Space Mapping

Based on the table above, some of the sample decision flow charts that can be constructed to utilize the problem-to-solution space mapping for PE designers are as follows:

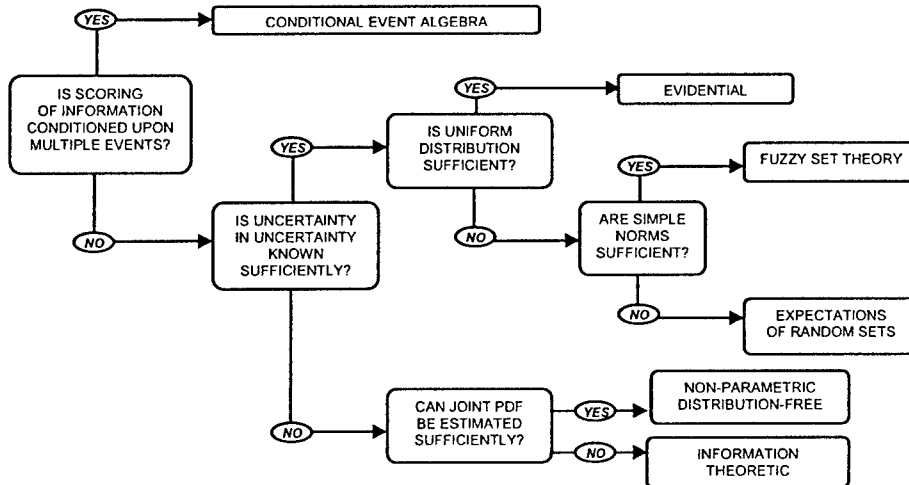
Example 1: Top-Level Hypothesis Evaluation Technique Selection



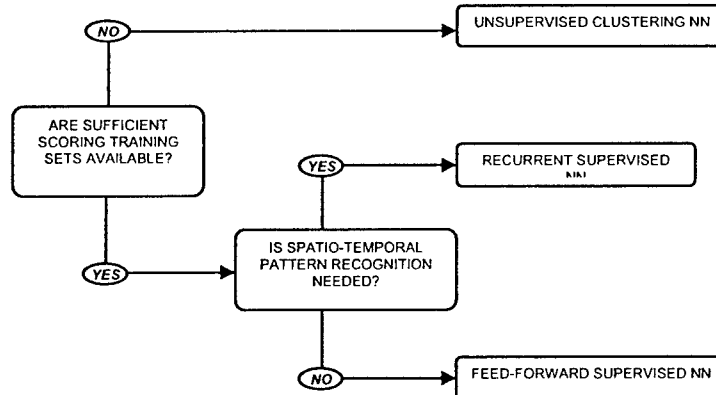
Example 2: Probabilistic Hypothesis Evaluation Technique Selection



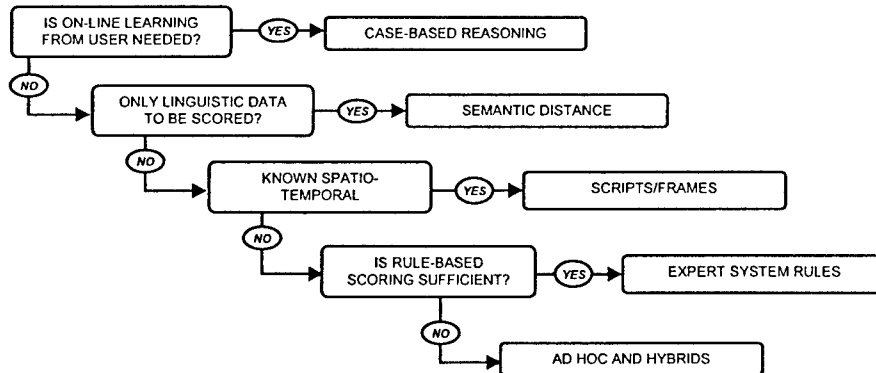
Example 3: Possibilistic, Non-Parametric and other Rigorous Hypothesis Evaluation Technique Selection



Example 4: Neural Network Hypothesis Evaluation Technique Selection



Example 5: Logical, Symbolic and *ad hoc* Hypothesis Evaluation Technique



3.3 Association Hypothesis Selection Problem-to-Solution Space Mappings

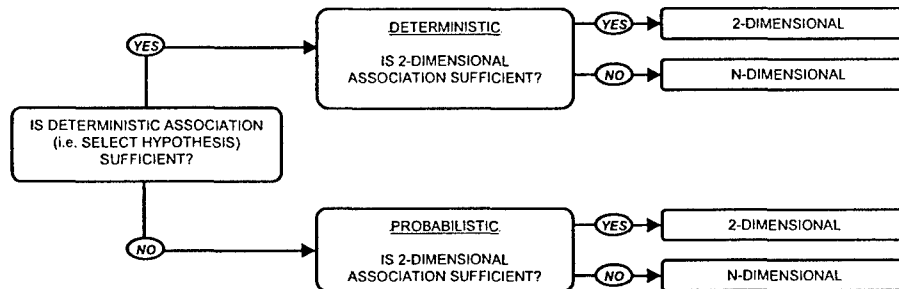
Figure 3-6 shows a sample problem-to-solution mapping for hypothesis selection.

| SOLUTION SPACE PROBLEM SPACE | Solution Type | | | | | |
|---|---|--|---------------------------|--------------------------------|--------------------------------|-------------------------------|
| | Probabilistic Set Partitioning Formulations | Hybrid: Probabilistic/Deterministic Set Covering | Set-Covering Formulations | Deterministic, 2D Formulations | Deterministic, ND Formulations | Approximations and Heuristics |
| BASIC PROBLEM FORMULATION | | | | | | |
| <i>True Randomness of data:</i> | | | | | | |
| • Random clutter | Y | | | | | |
| • Partial | | Y | | | | |
| • Target-related ambiguous data distributions | | | Y | | | |
| <i>Sensor Resolution:</i> | | | | | | |
| • Uniformly high | | | | Y | Y | Y |
| • Irregular | | Y | Y | | | Y |
| QUALITY AND SPEED OF SOLUTIONS FACTORS | | | | | | |
| <i>Accuracy, reliability of HF scores:</i> | | | | | | |
| • Low accuracy, reliability | | | | Y | Y | Y |
| • High accuracy, reliability | | | | Y | Y | |
| • High volume data | | | | | Y | |
| • Many (est.) true targets | | | | | Y | |
| • High quality (but local) assignments | | | | Y | | |
| • Reasonable quality (but "global") assignments | | | | | Y | |
| • Rapidly-computed solution | | | | Y | | Y |
| • Low complexity | | | | Y | | Y |
| STRUCTURE OF ASSIGNMENT MATRIX (2D) | | | | | | |
| • Very sparse | | | | Y | Y | Y |
| • Square | | | | Y | Y | |
| • Rectangular | Y | | | Y | Y | |
| • Real-valued cost | Y | | | Y | Y | |

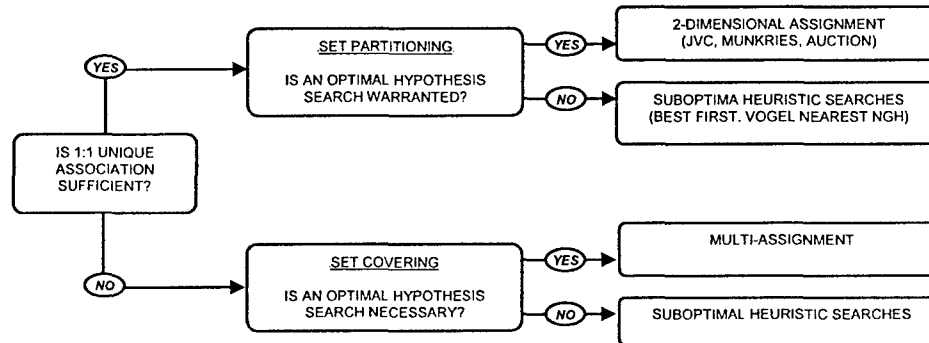
Figure 3-6: Hypothesis Selection Problem-to-Solution Space Mapping

Some examples of decision flow charts for Hypothesis Selection techniques are as follows:

Example 1: Top-Level Hypothesis Selection Technique Selection



Example 2: 2D Hypothesis Selection Technique Selection



Acknowledgements

This report reflects a research effort that would not have been possible without the support of both AFOSR and AFFTC, and that support is gratefully acknowledged herein, especially the programmatic support of Dr. Neal Glassman of AFOSR, and the technical guidance of Mr. Pete Burke and Mr. Martin Welch of AFFTC. This research was directly dependant on the use of simulator called CASE_ATT1, graciously provided by Defense Research and Development Canada (DRTDC/Valcartier) of the Canadian Department of National Defense. We also acknowledge the specific technical support of Mr. Nicolas Duclos-Hindié regarding CASE_ATT1 code and operations, and of Dr. Éloi Bossé of DRDC/Valcartier who authorized the use of CASE_ATT1 for this AFOSR project.

References

- [1] Llinas, J., Chen, L., et al, "Research on Range-Based Testing and Evaluation of Multi-Platform Air Force Teams", Center for Multisource Information Fusion, Report , Aug 24, 1999.
- [2] Llinas, J, Talreja, D., and Bowman, C. (2004). Design of a Performance Evaluation Methodology for Data Fusion-based Multiple Target Tracking Systems: Part 2: Analysis Methodologies. NSSDF JHAPL.
- [3] Rawat, S., Llinas, J., and Bowman, C. (2003) Design of a Performance Evaluation Methodology for Data Fusion-based Multiple Target Tracking Systems", presented at the SPIE Aerosense Conference, Orlando, FL, USA.

- [4] Blasch, E.P. (2000). Assembling an Information-fused Human-Computer Cognitive Decision Making Tool. IEEE Aerospace and Electronic Systems Magazine. Pp. 11-17.
- [5] Blasch, E.P. 2000. SAR Image Exploitation Using an Information Filter Metric. International Conference on Imaging Science, Systems, and Technology. Las Vegas, NV.
- [6] Blasch, E.P. and Schmitz, J. (2002). Uncertainty Issues in Data Fusion. National Symposium on Sensor and Data Fusion
- [7] Bennett, B., Bowman C., Lawrence, and Marcus, (2001). Performance Evaluation Strategies for Distributed Multi-Platform/Multi-Sensor Control Systems. MSS 2001 NSSDF Conference Proceedings
- [8] Drummond, O. E., and Fridling, B.E. "Ambiguities in Evaluating Performance of Multiple Target Tracking Algorithm," Signal and Data Processing of Small Targets 1992, Proc. SPIE, Vol. 1698.
- [9] Martin Hofmann and Steve Jameson, "Complexities and Performance Assessment for Data Fusion Systems", Proceedings of the 1998 IRIS National Symposium on Sensor and Data Fusion. March 1998.
- [10] Bowman, C., (2004). The Dual Node Network (DNN) Data Fusion & Resource Management (DF&RM) Architecture. AIAA Intelligent Systems Conference, Chicago, IL, USA.
- [11] Davey, D.J. (2003). Extensions to the Probabilistic Multi-Hypothesis Tracker for Improved Data Association. PhD Dissertation. University at Adelaide, Australia.
- [12] Ross, T.D., Bradley, J, Hudson, L., and O'Conner, M. (1999). SAR ATR - So What's The Problem? - An MSTAR Perspective, SPIE Proceedings. Vol. 3721.
- [13] Blasch, E. and Plano, S. (2002). JDL Level 5 Fusion Model 'User Refinement' Issues and Applications in Group Tracking. SPIE Proceedings. Vol. 4729. pp. 270-279.
- [14] Blasch, E., Pribilski, M., Daugherty, B., Roscoe, B., and Gunsett, J. (2004). Fusion Metrics for Dynamic Situation Analysis. SPIE. Orlando, FL, USA.
- [15] Bar-Shalom, Y. and Tse, E. (1975). Tracking in a cluttered environment with probabilistic data association, Automatica 11.

Performance Evaluation Methods for Network Centric Data-Fusion Capable Tactical Platforms

Kedar Sambhoos, Satyaki Ghosh Dastidar, James Llinas

Center for Multisource Information Fusion

University at Buffalo,

Buffalo, NY-14260, U.S.A.

kps6_sg44@buffalo.edu, llinas@eng.buffalo.edu

Christopher Bowman

Data Fusion & Neural Networks

1643 Hemlock Way,

Broomfield, CO – 80020, U.S.A.

cbowmanphd@msn.com

Abstract - Network centric warfare poses new challenges and opportunities to the fusion community. In a network centric environment, fusion technology impacts the ability to consume, create or act on information through proper allocation and utilization of available resources, and supports awareness, impact assessment, and action. While technological advances continue to take place in multisource data fusion (DF), a proper performance evaluation (PE) methodology is required to evaluate performance of alternative DF process designs that are being developed to handle the increasing complexity of modern day applications. PE processes need to be technically fair yet affordable to evaluate various data fusion (DF) measures of performance (MOP). This paper addresses the distributed fusion problem and gives quantitative insights into the interdependencies of fusion processes and the consistency measures between distributed fusion products. Building on our prior works, our recommended PE methodology is based upon the Dual Node Network (DNN) DF & Resource Management (DF&RM) architecture. Our case studies involve track picture consistency across multiple platforms and sensors for what we label as Tier 1, Tier 2 and Tier 3 Level 1 fusion (i.e. entity or object assessment). The highlight of the paper is a design of experiments (DOE) framework from which we identify the effects and interactions of various MOPs (factors). We also propose a response optimization method to adjust the factor parameters for best possible track picture consistency. This research focuses on distributed Level 1 DF PE applications for the Air Force Flight Test Center (AFFTC), in support of new test and evaluation procedures that will be required for advanced, fusion-capable tactical aircraft.

Keywords: Performance Evaluation, Distributed Data Fusion, Dual Node Network Architecture, Measures of Performance, Design of Experiments.

1 Introduction

Linn, Hall and Llinas [8] describe over 50 prototypical systems for multi-sensor data fusion systems that have been developed for Department of Defense (DoD)

applications some ten years ago. Such systems have become ever more sophisticated. Many of the prototypical systems summarized by Linn *et al.* [8] utilize advanced target identification techniques. While much research is being performed in the data fusion community to develop and apply new algorithms and techniques, much less work has been done to determine how well such methods work. In the context of target tracking and estimation, Multi-Sensor Data Fusion (MSDF) is used to combine data from redundant and/or complementary sensors, to generate complete and precise information regarding location and identity of unknown numbers of unknown targets of different types. In most cases it is not possible to deduce a comprehensive picture about the entire target scenario from each of the pieces of evidence alone, due to the inherent limitations of technical features characterizing each sensor. Judicious trade-offs between computational complexity, computational time and numerical accuracy have to be made for selection of an algorithm for practical applications; such tradeoffs can imply large economic impact if the evaluation is associated with a contract competition, or more seriously can affect life-and-death decisions if the fusion products are used for decision support.

The employment of rigorous, consistent, and equitable Performance Evaluation (PE) methods for data fusion processes contribute to the probability of success when that system is employed on an operational mission. An extendable framework for PE of distributed fusion and response management software is needed to (i) stop building one-of-a-kind PE systems, (ii) expose alternative PE designs to handle the increasing complexity of the test articles, and (iii) provide an affordable, yet equitable, evaluation of alternative Data Fusion and Response Management (DF&RM) systems. There is a need for an extensible framework for PE that encapsulates all known approaches for the variety of PE problems.

2 Objectives

This paper describes the PE research and the related software development that was done as a part of the ongoing research for the Air Force Flight Test Center

(AFFTC), conducted under support from the Air Force Office of Scientific Research (AFOSR). The previous work by Ghosh Dastidar, Sambhoos, Bowman and Llinas [5] extended the formalized PE methodology developed for Level 1 tracking-based MSDF systems (in [2] and [4]). We included a summary of the proposed PE methodology herein, although our focus in the current work is on the issue of PE for inter-platform track picture consistency, as part of our efforts to begin extending the proposed PE framework to the case of *distributed* MSDF. In all of our efforts, because of the special interests of AFFTC, we have focused on MSDF applications typical of tactical aircraft systems.

The earlier paper [5] gave quantitative insights into the interdependencies between distributed fusion measures of performance (MoPs) and (i) track-to-truth association for accuracy, and (ii) track-to-track association for consistency. The goal of this paper is to extend the PE software capability to simulate and evaluate performance metrics for distributed fusion test articles combining distributed data from network centric type communication with different types of sensors (Radar, ESM,IRST) with different data fusion management (ownership, cooperative) nodes. Figure 1 illustrates such a network centric communication architecture. A simulation-based, case-study approach along with a statistically-rigorous Design of Experiments (DOE) framework is employed as the basis to explore our methodological ideas for PE as applied to Level 1

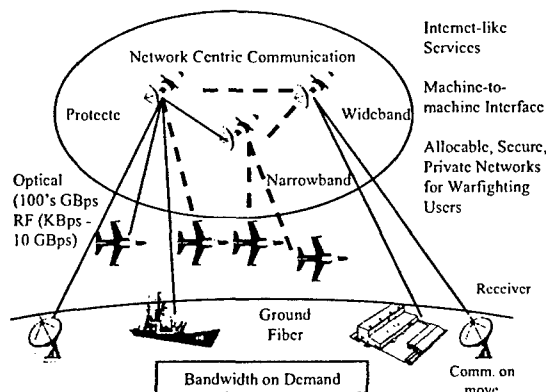


Figure 1 : Network Centric Warfare Communication. MSDF.

The remainder of the paper is organized as follows. Section 3 discusses the PE framework methodology. Section 4 introduces and discusses the various PE metrics. Section 5 describes the PE software requirements and Section 6 describes the case study simulator for the associated PE network architecture in the context of the requirements of AFFTC. Section 7 explains the Tier based PE node design. Section 8 discusses the experimental results from the case study and Section 9 describes the DOE framework and results.

Finally Section 10 presents a summary of this research and the future directions to this research.

3 PE Framework¹ Development

A central issue in evaluating any prototype data fusion process (here, fusion-based tracking) is the problem of determining which fused estimate output by the prototype fusion test article should be compared with which "truth" state (here, truth track or other fused track providing the basis of the assessment)². The importance of addressing this issue is based on the assumption that errors in fusion-produced state estimates will be computed by comparing estimates to truth states. Thus, to compute estimates for metrics of interest, the association between the estimates and truth must be established.

According to Roy and Bosse [11], there are three broad issues that lead to the ambiguities in Track-to-Truth. These are (i) *Mis-association Issues*, (ii) *Track Management Issues*, and (iii) *Tactical Picture Issues*. The "Track-to-Truth" association problem is one inherent difficulty in evaluating any MSDF process. In our prior works, we have employed Drummond's ideas on a couple of ways that the PE process could account for this issue (see [9] and [10]).

3.1 Need for a PE Framework

A typical data fusion process that shows the role of runtime performance assessment/performance evaluation is shown in Figure 2. An effective PE network is needed to expose the differences between alternative approaches

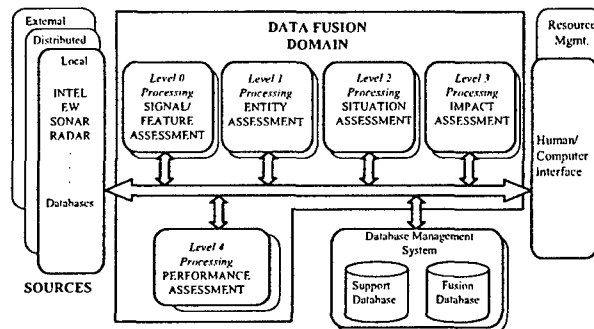


Figure 2: PE as a Level 4 fusion function.

and to organize the alternative DF&RM Test and

¹. According to IEEE STD 610.12 definition [7] and the DoD Architecture Framework Working Group, a *framework* or an *architecture* is a structure of components, their relationships, and the principles and guidelines governing their design and evolution over time.

². This issue carries over to any fusion state estimate, not only kinematic tracking. Note too that it is assumed that the truth state is somehow known; in digital simulation-based testing this is typically straightforward but in other test settings determining the truth state may not be so easy itself.

Evaluation (T&E) software design patterns The U.S. Air Force needs effective PE systems for their Systems Under Test (SUTs) at AFFTC to evaluate cost/complexity versus PE performance relationships between PE methodological choices and their effects on metric computation.

The PE system cost/complexity versus PE performance relationships between PE methodological choices and their effects on metric computation cannot be fully developed without this framework. The PE framework needs to be applicable to multi-level DF&RM software T&E. Our recommended approach is that, because of the need to associate estimates-to-truth, design of the PE process entails the design of a *new data fusion process* specific to the satisfaction of PE requirements. Thus, PE is treated herein as a fusion process as defined in Steinberg and Bowman [12]; while there is a challenge in designing a PE process as a fusion process, this situation does allow the exploitation of the existing data fusion technology knowledge base in understanding PE problem solutions. By treating PE as a fusion process the DF&RM DNN architecture [12] provides a baseline for the PE framework. This enables all the techniques that exist for all the levels of fusion to be considered for each corresponding PE function. Steinberg and Bowman [12] approach PE as a Level 4 fusion function as shown in Figure 2.

3.2 Criteria for a PE Framework

The PE framework needs to provide standard components, interfaces, and guidelines that enable software reuse and extendibility to achieve affordability objectives. The PE framework needs to expose alternative solutions with established component interfaces to permit comparison, integration, and interoperability objectives. The PE framework should help achieve reduced cost of development by promoting expandability, modularity, and reusability of its PE solutions. PE system design criteria include mission measures of effectiveness (MME), measures of effectiveness (MoE), and measures of performance (MoP) accuracies, especially for those measures that distinguish the performance of candidate SUTs.

4 Comprehensive Performance Evaluation Metrics

The employment of a comprehensive PE approach would yield both measures of the effectiveness and performance of a fusion system. Additionally, the PE network and its nodes must be designed to achieve *fairness* in PE system performance versus cost/complexity. The PE design objective is to generate fair (e.g., accurate) DF system mission EEI metrics (e.g., MME, MSEs, MoPs) with minimal cost. As such the PE MoPs need to be computed with sufficient accuracy to

differentiate SUT performance with respect to the scenario MoPs which are driven by the mission objectives, scenarios, and the SUTs. Steinberg and Bowman [12] mention some different measures and the associated metrics for a fusion system. The PE measures fall into three categories: (i) Mission related (MMEs), (ii) Operational (MoEs) and (iii) Engineering (MoPs). There should be a traceable interconnectedness among these measures. The mission measures are the top-level measures

The canonical MME is the overall probability of mission success. Classic MoEs for fusion and management test articles include those that measure, e.g.:

- The nature of enemy behavior more completely, more efficiently, more accurately, more quickly, over a wider area, and without being detected.
- The impact of a strike more efficiently, more accurately, more quickly, over a wider area, and without being detected.

The MoPs support the MoEs by providing specific performance insights. Traditional MoPs are location and ID accuracy and probability of detection and false track. Refinements of Level 1 fusion metrics are used to provide additional insights into their corresponding MoPs. Data association MoP refinements may include:

- *Track Purity (Targets/Track)*: – Ratio of track segments in an integrated track that belongs to same target (or group of targets (N_{TS}), to total number of segments in a track (N_{TST}). ($T_P = N_{TS} / N_{TST}$).
- *Track Fragmentation (Tracks/Target)*: the number of hypotheses to which elements of an actual aggregation are assigned as elements by the system.
- *Hypothesis Proliferation (Tracks/Report)*: the number of competing (overlapping) tracks per report.
- *Assignable Track Ratio*: Fraction of the tracks that are associated with exactly one target.
- *Non-Assigned Target Ratio*: Fraction of the targets to which no tracks are assignable.

5 Fusion PE Software Development

5.1 Design of the PE Fusion Process

It is imperative to develop good performance evaluation software to calculate, study, and analyze PE metrics. Broadly speaking, PE receives the tactical picture output from the distributed fusion test articles, (i.e., from the fusion nodes in the DNN of the SUT). PE receives the truth from the simulation, commands from the user, and support services as directed by the user. PE outputs Measures of Performance (MoP) results and generates displays for the analyst. Performance evaluation software associates track-to-truth and estimates the MoPs. Thus PE is type of fusion problem where truth and the Consistent Track Picture (CTP) track files are associated and MoP state estimates are

generated based upon this association. Thus PE software design can be performed with the same architecture as used for fusion, as noted previously. Rawat *et al.* [9] and Bowman [1] discuss the issues of designing the PE process as a fusion process and discuss the development of such an architecture from the (i) role optimization phase, to the (ii) network optimization phase, (iii) to the node optimization phase, and finally into the (iv) software pattern optimization phase. Figure 3 shows the entire process with the requirements, design, and evaluation refinement step for each phase.

5.2 PE Node Optimization

Figure 4 shows the functions for any performance evaluation node in PE process architecture. The various steps involved are as follows:

1. *PE Data Preparation*: This performs track and truth common reference transformations. It accounts for known spatial and temporal misalignments
2. *PE Data Association*: This performs track-to-truth or track-to-track association using kinematics, attributes and ID of the distributed track files.

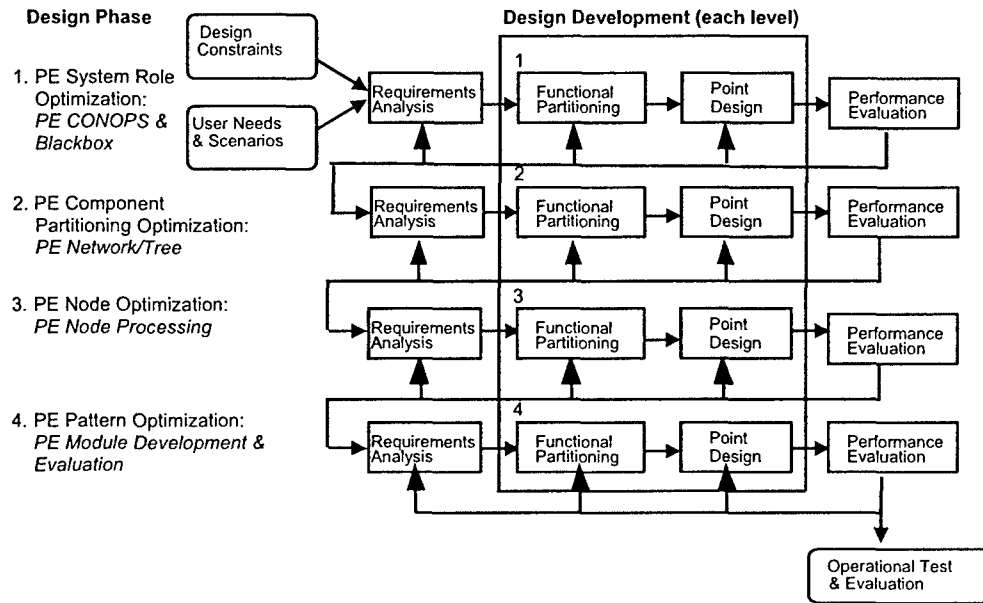


Figure 3: Different design phases of the PE fusion process.

In the context of AFFTC test and evaluation (T&E) framework, PE provides (i) performance evaluation of the different DF&RM test articles and (ii) utilizes the test management and support services.

3. *PE State Estimation*: This estimates the MoPs using selected associations (e.g., deterministic, MHT, probabilistic, Wasserstein).

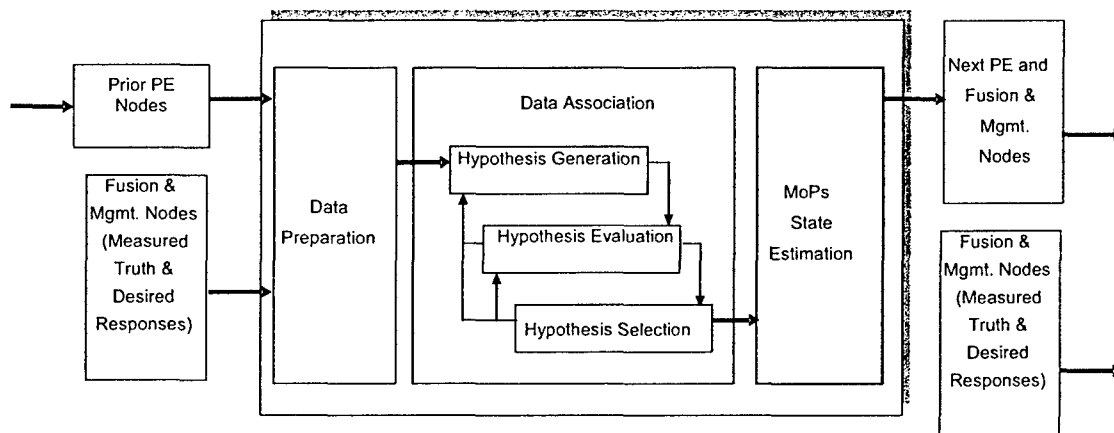


Figure 4: PE node components according to DF&RM Dual Node Network (DNN) architecture.

6 Case Study: PE Simulator for AFFTC

In the above, we have summarized some of the generalized issues when considering the test and evaluation of a prototype data fusion process (what we have called the "System Under Test" or SUT). The current research is focused on the problem of PE and the "fairness" issue for the distributed data fusion case. In future AFFTC applications, one type of expected distributed fusion application will be the case of multiple aircraft platforms working cooperatively on a common

Study, during data preparation the PE node puts tracks and truth information in $[x, y]$ co-ordinates and common time. Data association performs deterministic track-to-truth association and track-to-track association. During data association the following three actions are performed:

- (i) Hypothesis Generation,
- (ii) Hypothesis Evaluation, and
- (iii) Hypothesis Selection.

The PE node uses a Kalman filter for MoP state estimation. Figure 5 shows the generic PE node network

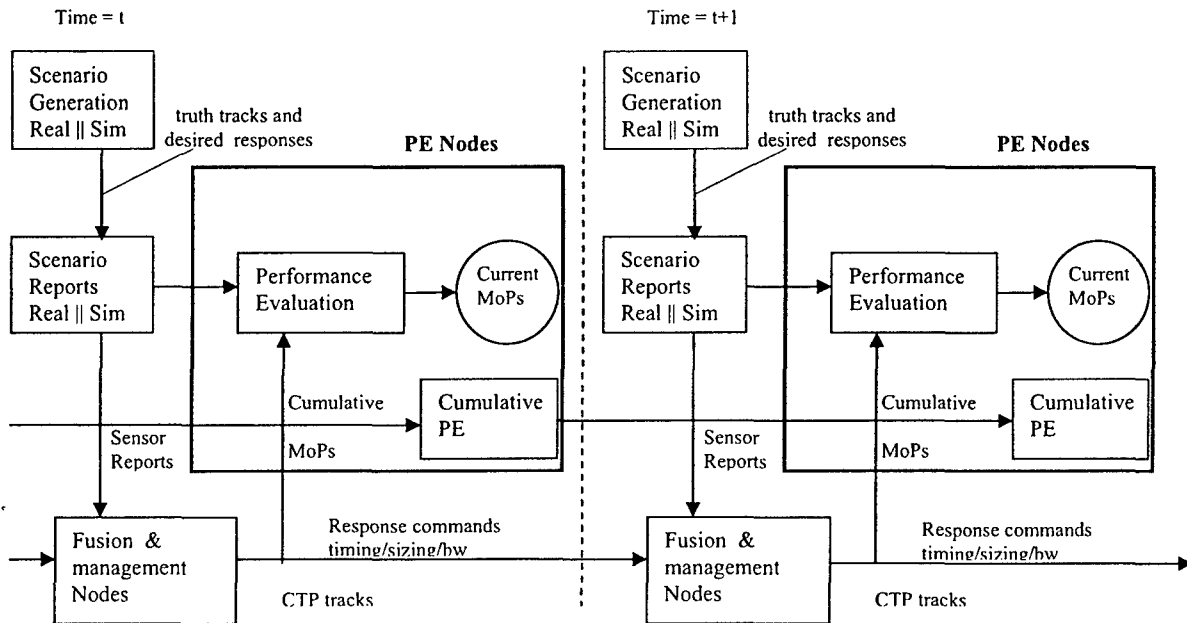


Figure 5: Generic PE node network (On or Off Line).

mission, each performing local or platform-specific fusion while also exchanging data and fused estimates to each other. A core evaluation issue herein is the assessment of the degree of consistency in the multiple track pictures across the platforms. It is also critical to guarantee that the alternative PE network node outputs are consistent, in accordance with a consistency specification pertinent to the application. PE nodes perform track-to-truth association to support track accuracy-related or other MoPs, and perform track-to-track association to support platform track file consistency-related MoP estimation for two or more intermetted platforms (e.g., Joint Strike Fighters (JSFs), or the "F-35" aircraft).

6.1 PE Node Design

In the PE framework the PE nodes perform 3 necessary functions: (i) data preparation (ii) data association and (iii) MoP state estimation. In our Case

for online and offline scenarios.

6.2 Case Study Measures of Performance for PE

Figure 6 depicts how the two platforms have their own view of the truth picture based on the on-board sensors. There are both "common" pictures and "unique" pictures. Let us assume, for the sake of example, that all the on-board sensors see the same targets. Let platform 1 sees 3 tracks (based on on-board sensors) which are common to platform 2 and vice versa. The common tracks are shown in red. Note that even though both of the platforms see the same targets, their measurements about those common targets could be different depending on how the on-board sensors reports the measurements. Also there are certain targets that are uniquely seen by platform 1 and platform 2; note that some of either the common or unique tracks could be

false tracks.

Each of the platforms exchange their track files and

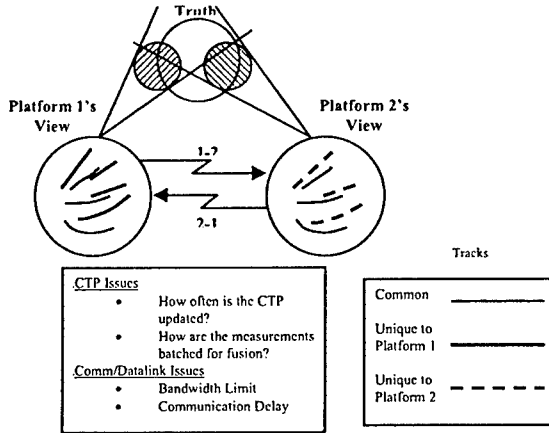


Figure 6: Track pictures of individual platforms.

data fusion is done upon receipt of this information at each platform. We will explain further how this information is exchanged when we discuss Tier 0, Tier 1 and Tier 2 (Section 6.3). We assume that there is no bandwidth limitation in communication. An improvement from the earlier work is that we have incorporated more realistic asynchronous (delayed) communication among the sensors and the platforms.

The baseline distributed fusion output is the Consistent Tactical Picture (CTP). The sensor track file "consistency" is computed at each time point as the percentage of matching CTP tracks in the track files of each platform. In addition to this measure, the following four consistency metrics have been computed:

1. *Track-to-Track Consistency*: The percentage of matching tracks across all the platforms' track files w.r.t. the average number of track files per platform that are generated over the scenario. For example, if one platform currently has 7 tracks and the other 9 with 6 associated (i.e., matching within bounds), then the average consistency is $6 / \{[7+9]/2\} = 0.75 = 75\%$.
2. *Track-to-Truth Consistency*: The percentage of average number of matching tracks per platform w.r.t. true number of tracks. For example, suppose one platform has 7 tracks with 3 associated tracks and the other has 10 with 4 associated tracks. Let the number of true tracks be 5. Then the average track to truth consistency is $\{3/5 + 4/5\} / 2 = 0.7 = 70\%$.
3. The average number of standard deviations of error in the associated tracks at each time point

$$= \frac{\sum_{i=1}^K (I^T V^{-1} I)^{\frac{1}{2}}}{K}$$

and the average is over the scenario where

- I is the 1×2 matrix difference of feasibly

associated track state 2-vectors (i.e., $Y(track_i) - Y(track_j)$ in x and y position) between any two sensors i and j .

- $V = P_1 + P_2$ is the covariance of the error which here is the sum of the track state error covariances in x and y positions for each selected association
 - K is the total number of selected associations used in the consistency score.
4. The average location error standard deviation of associated tracks at each time point

The above equation is the association score that must be greater than the non-association threshold given by Equation (3). PE receives the fused tactical picture output from the distributed fusion test articles. PE receives the truth from the simulation, commands from the user, and support services as directed by the user. PE generates MoP results and generates displays for the analyst. After each platform's track file is associated, the consistency performance measures over the entire scenario are computed.

6.3 Explanation of Level 1 Fusion Tiers

6.3.1 Tier 0

In Tier 0, each of the on-board sensors (Radar, ESM and IRST) fuse their own reports. The resultant Tier 0 tracks are then fused together to get the Tier 1 consistent track picture. Here the information is not yet shared across the platforms, so the result tends to be less accurate than for example the fusion of Tier 0 sensor tracks to the all source CYP. Generally, batching of larger data sets for fusion is more accurate; albeit more complex.

6.3.2 Tier 1

In Tier 1, each of the on-board sensors (Radar, ESM and IRST) share their Tier 0 track files to generate the ownship consistent track picture. This is typically done for each sensor track file as it is updated, rather than all sensors at once. The DNN architecture exposes these and many other ways to network fusion nodes on a single platform for Tier 1 fusion or on multiple platforms for Tier 2 fusion.

6.3.3 Tier 2

In a typical Tier 2 fusion the Tier 1 track files are fused sequentially as each Tier 1 track file is updated. A modified form of a Tier 2 fusion network is for each platform to share its own sensor measurements with the other platforms. This can be done one sensor at a time sequentially as each sensor scan of data is received. This alternative tends to be more accurate, however at a cost of more communications bandwidth and fusion complexity (e.g., due to report propagations for time

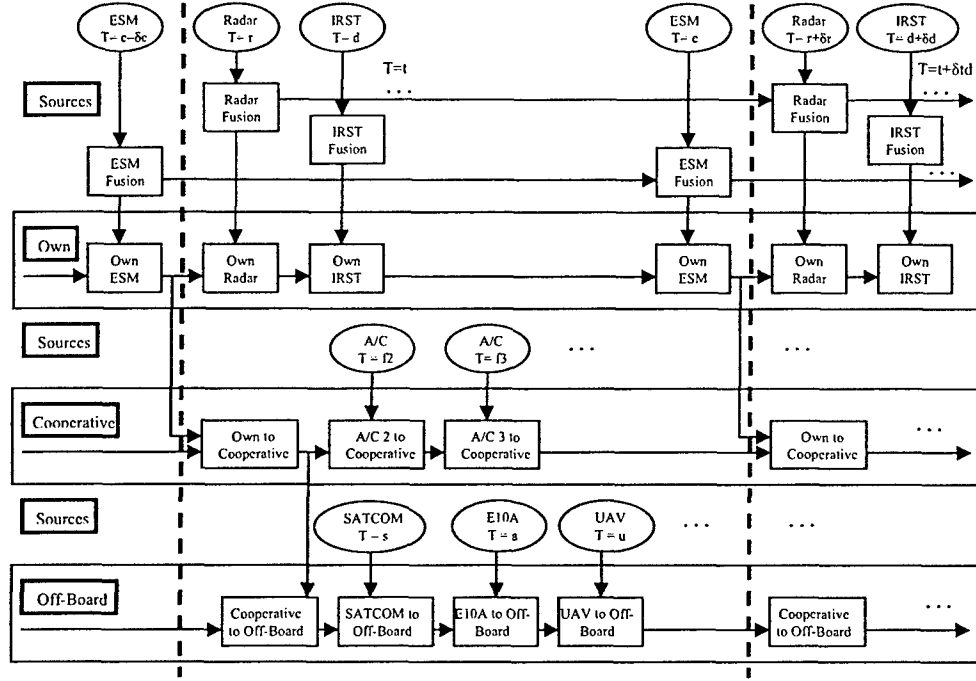


Figure 7: Baseline Level 1 Fusion Network for Tiers 0, 1 and 2.

delays, multiple platform coordinate misalignments, internetted ghost tracks, etc.).

6.4 Baseline Level 1 SUT Fusion Network

The Level 1 fusion network involves entity state estimation based upon data shared among the sensors (Radar, ESM, IRST) and the sources (e.g., aircraft, satellite, E10A, UAV) on each ownship, cooperatively, and offboard sources. The Level 1 fusion node network includes Tier 0 sensor report fusion nodes sequenced over time batches of reports from each sensor. The Tier 0 associated reports and tracks are fused in a sequence of Tier 1 ownship fusion nodes over sensors and time. The ownship fused results from each platform are fused in a sequence of Tier 2 cooperative fusion nodes. Figure 7 illustrates the network architecture.

7 Tier Based Fusion Node Design

The Tier 0, 1 and 2 fusion node detailed designs are presented for (i) Tier 0 report-to-track sensor fusion, (ii) Tier 1 track-to-track ownship fusion, and (iii) Tier 2 track-to-track cooperative fusion

7.1 Coordinate Conversion

The spherical co-ordinates. (R, Az, El) from the sensor reports are converted to (E, N, Up) co-ordinates. The reports are converted from (R, Az, El), with azimuth clockwise from the projection of the velocity to level and elevation up from level, to the (E, N, Up) relative platform (R, Az, El), with azimuth measured clockwise

from north and elevation up from local level using a flat earth. These (R, Az, El) reports are converted to a fixed centered (E, N, Up) Cartesian coordinate system for use by the Tier 0 fusion SUTs. In this paper, we only used the [E, N] as the [x, y] coordinates. RAE position and covariance inputs are converted to the ownship centered E, N, Up (ENU) coordinate system using range, r, Azimuth from N, Az, and elevation from level, El in the reports as follows:

$$E = r \sin(Az) \cos(El)$$

$$N = r \cos(Az) \cos(El)$$

$$Up = r \sin(El)$$

The range rate conversion to east and north rates is as follows:

$$E^* = R^* \sin(Az) \cos(El) + R \cos(Az) Az^* \cos(El) - R \sin(Az) \sin(El) El^* = R^* \sin(Az) \cos(El)$$

$$N^* = R^* \cos(Az) \cos(El) - R \sin(Az) Az^* - R \cos(Az) \sin(El) El^* = R^* \cos(Az) \cos(El)$$

$$Up^* = R^* \sin(El) + R \cos(El) El^* = R^* \sin(El)$$

where

E^* is east rate, N^* is north rate,

R^* is the range rate measurement, and Az is azimuth measurement,

$Az^*=0$ and $El^*=0$, since azimuth rate and elevation rate are not measured and the expected value of zero is used.

For sensor tracks the 3x3 top left corner of the ENU error covariance matrix, $R(3 \times 3)$ of the whole $R(6 \times 6)$, is rotated by the report azimuth and elevation from the sensor R_0 RAE error covariance oriented along north and level as follows:

$$R(3 \times 3) = \Phi R_0 \Phi^T$$

where R_0 is the independent measurement error covariance and its diagonal elements are the sensor azimuth, range, and elevation error variances in meters such as for the baseline radar:

$$\begin{bmatrix} (8 \text{ mrad} \cdot \text{range})^2 \text{ m}^2 & 0 & 0 \\ 0 & (25)^2 \text{ m}^2 & 0 \\ 0 & 0 & (16 \text{ mrad} \cdot \text{range})^2 \text{ m}^2 \end{bmatrix}$$

where 'range' is the sensor range measurement.

An example for the ESM in track mode for the baseline (Az, R, El) error covariance is

$$\begin{bmatrix} (8 \text{ mrad} \cdot \text{range})^2 \text{ m}^2 & 0 & 0 \\ 0 & (0.4 \cdot \text{range})^2 \text{ m}^2 & 0 \\ 0 & 0 & (16 \text{ mrad} \cdot \text{range})^2 \text{ m}^2 \end{bmatrix}$$

where 'range' is twice the sensor range measurement with a cap at the max detection range and a floor at $\frac{1}{4}$ the max detection range.

Φ is the 3x3 rotation matrix (i.e., rotation by report azimuth about up clockwise and then rotation by report elevation about the axis orthogonal with the azimuth and up plane) defined as follows:

$$\begin{bmatrix} \cos(\text{Az}) \sin(\text{Az}) \cos(\text{El}) & -\sin(\text{Az}) \sin(\text{El}) \\ -\sin(\text{Az}) \cos(\text{Az}) \cos(\text{El}) & -\cos(\text{Az}) \sin(\text{El}) \\ 0 & \sin(\text{El}) \cos(\text{El}) \end{bmatrix}$$

For the radar, the 3x3 lower right corner of the radar ENU error covariance matrix, $R^*(3 \times 3)$ of the whole radar (E, N, Up) error covariance $R(6 \times 6)$, is computed by rotating by the sensor R_i rate error covariance (where azimuth rate from north is row 1, range rate is row 2, and elevation rate from level is row 3), by the Φ 3x3 rotation matrix as follows:

$$R^*(3 \times 3) = \Phi(3 \times 3) R_i(3 \times 3) \Phi^T(3 \times 3)$$

In this paper, $R_i(3 \times 3)$ is the independent rate measurement error covariance with its user specified diagonal elements as follows:

- Azimuth rate aircraft dynamics constraints (e.g., Gaussian speed variations with a 3 sigma of about 900 m/sec),
- Sensor range rate measurement variance in (meters/sec)², and
- Elevation rate aircraft dynamics constraints (e.g., Gaussian climb/dive variations with a 3 sigma of about 150 m/sec).

This baseline R_i^* matrix then becomes

$$\begin{bmatrix} 300^2 (\text{m/sec})^2 & 0 & 0 \\ 0 & 1^2 (\text{meters/sec})^2 & 0 \\ 0 & 0 & 50^2 (\text{m/sec})^2 \end{bmatrix}$$

The resulting sensor measurement error covariance matrix, R , is the 6x6 block diagonal matrix with $R(3 \times 3)$ in upper left and $R^*(3 \times 3)$ in lower right and zeros elsewhere.

The above results in report state and its error covariance in fighter centered (E, N, Up) coordinates. To transform these to fixed coordinates at each time point use the fighter position as given by its inertial navigation system (INS) to translate to fixed (E, N, Up) coordinates. Thus the sensor report is translated using the ENU fighter coordinates, C_k , at the time, k , of the report as follows:

$$Y^+_{\cdot k}(T_k) = Y^+_{\cdot k}(T) + C_k$$

where Y is the full (E, N, Up) position and velocity vector for the sensor report. For this design no time alignment is needed.

7.2 Tier 0 Sensor Fusion Node Design

The sensor report-to-track fusion node processing from the previous effort [5] was used as the starting point for each of the 3 sensors. The report and track kinematics are in earth fixed (E, N) coordinates. Since we chose constant elevation, we ignored the Up axis.

Data Preparation propagates the previous sensor tracks to the expected next-measurement time for data association which then generates, evaluates, and selects from the alternative report-to-track associations versus track initiation or deletion hypotheses. State estimation then updates the sensor track file based upon the hypothesis selection.

7.2.1 Data Preparation

The primary sensor data preparation design operation is track file propagation. All sensor tracks are propagated to the current sensor report time. This is done for the sensor states and their covariance matrices via multiplication by a time dependent phi matrix and the addition of noise for the uncertainty in the entity dynamics and aircraft navigation error over this period of time. Namely, the sensor track state, x , and its covariance, P , is propagated (e.g., x seconds forward) to the current report state time as follows:

$$\begin{aligned} Y^+_{\cdot k}(T) &= \phi(\delta t) Y^+_{\cdot k-1}(T), \text{ and} \\ P^+_{\cdot k} &= \phi(\delta t) P_{k-1} \phi^T(\delta t) + Q(\delta t). \end{aligned}$$

where,

- $Y^+_{\cdot k-1}(T)$ is the best estimate at time $k-1$ of the sensor track T kinematics state which consists of position and velocity in (E,N,Up); $Y^+_{\cdot k-1}(T)$ is actually a 6-d column 6x1 vector of (x, y, z, v_x, v_y, v_z) where x is East, y is North and z is Up.
- $Y^+_{\cdot k}(T)$ is the estimate of the ownship track T kinematics state propagated from time $k-1$ to time k and before an update at time k .
- $\phi(\delta t)$ is the 6x6 state transition matrix for a change in time of δt from time $k-1$ to time k for each dimension of a sensor kinematics state. This assumes a constant velocity model in each axis, (i.e., ϕ is the identity with δt 's along the block diagonal of the upper right 3x3;
- P_{k-1} is the error covariance for $Y^+_{\cdot k-1}$ at time $k-1$. The functional T dependence is suppressed since it would appear for all covariances P of sensor tracks, T .
- $P^+_{\cdot k}$ is the error covariance for $Y^+_{\cdot k}$ at time k before an update is performed in the sensor track. $P^+_{\cdot k}$ is a 6x6 matrix.

- $Q(\delta t)$ is the white noise process uncertainty 6×6 covariance matrix over a time difference of δt for the kinematics model for each axis, which is an input depending primarily upon the velocity uncertainty per entity type. A simple random walk model for velocity in each axis is used with the process noise in velocity with a specified variance (e.g., $0-250 \text{ m}^2/\text{sec}^2$ for vehicles, $0-2000 \text{ m}^2/\text{sec}^2$ for helicopters, and $0-1000 \text{ Km}^2/\text{sec}^2$ for fixed wing aircraft). Q is equal to this velocity variance, with a default value of $(300 \text{ m/sec})^2$, times the matrix $[\delta t^2, \delta t, \delta t, 1]$ in each axis. Namely, Q is the 6×6 symmetric positive definite matrix containing four 3×3 blocks. The upper left has δt^2 along the diagonal and the lower right has 1's along the diagonal. The other two have δt 's along their diagonals.

The resulting P_k^+ covariance needs to be increased by the ownship INS error covariance in the new location at time k , N_k , is added to the track position error covariance, P_k^+ , to yield the translated error covariance. The 6×6 N_k matrix is computed based upon the delta time, δt , from the last update of this track to the current report time. We assumed this error covariance, $N_k(\delta t)$, to be a constant with a baseline of $(5 \text{ m})^2$ in each position axis and $(0.1 \text{ m/sec})^2$ in each velocity axis and is added to the propagated P_k^+ , viz.,

$$P_k^+(T_k) = P_k^+ + N_k(\delta t)$$

7.2.2 Data Association

The primary sensor track to sensor report data association design operations are the following:

1. **Hypothesis Generation:** Find feasibly associated sensor tracks within gates of each sensor report [use nearest (in time) last updated sensor track propagated to each feasible sensor report time]
2. **Hypothesis Evaluation:** Compute MAP scores for all feasible hypotheses
3. **Hypothesis Selection:** Apply the existing fusion node SUT Vogel, best first, assignment algorithm to select the associations.

7.2.3 Hypothesis Generation

The baseline design uses the (E,N,Up) 3-D position-only chi-square gate as used in the existing SUT to avoid detrimental affects of inaccurate velocity estimates or inaccurate entity typing. The kinematics gate is parameterized (e.g., $M=25$ for a 5 sigma gate). Namely, for the report, $Y(S)$, gate out all sensor tracks, $Y(T)$, such that

$$I^T V^{-1} I > M$$

where

- I is the 3×1 innovations column vector,

$$I = Y(S) - H Y^+(T).$$

- H is the measurement to state conversion matrix that is the 3×6 conversion matrix to 3-D from track 6-D with identity in the first 3×3 and remainder zeros
- V is the innovations covariance,

$$V = H [P_k^+] H^T + R.$$
- $Y(S)$ is the 3×1 portion of the sensor report Gaussian kinematics with 3×3 error covariance R , which is the covariance of the position measurement error in E, N, Up coordinates.
- $Y(T)$ is the track Gaussian kinematics with covariance P_k^+ propagated to current time above in data preparation.
- M is the gate around $Y(S)$, outside of which $Y(T)$ is infeasible, for simplicity a 3-sigma gate is used.

In this paper, we used a 2-D-only position gating (i.e., not using altitude). If all gates are passed then the pair is marked as a feasible association (we presume that each sensor and sensor track is a track on a single entity). Each sensor track passing the gates is included in the feasibly associated tracks for each sensor report as in the existing SUT.

7.2.4 Hypothesis Evaluation

The baseline scoring equations with entity type classification network confidences are based upon the standard Max a Posteriori (MAP) hypothesis evaluation scoring defined as follows:

$$\begin{aligned} \max P(H|R) &= \max \{P(R|H) P(H)\} \\ &= \max \{P(Y|H) P(Z|Y,H) P(H)\} \\ &= \max [\Pi_T \{P(Y(S)|Y(T),H) P(Z(S), \\ &\quad Z(T)|Y(S), Y(T), H) P(H)\}] \end{aligned}$$

where

- the maximization's are over all association and non-association hypotheses, H ,
- H is the set of feasible association or non-association hypotheses,
- R represents both the sensor track, T , and sensor report, S , data,
- Y is the set of kinematics from both,
- Z is the set of all parameters & entity attributes measured in both
- the product is over all independent track, T , hypotheses (i.e., of all 5 types),
- $Y(T)$ are the track kinematics, the $P(Y(T)|H)$ term is dropped as constant with respect to the maximization,
- $Y(S)$ are the sensor report kinematics,
- $Z(T)$ are the parameters and entity attributes from the track,
- $Z(S)$ are the parameters and entity attributes from the sensor report,
- $P(H)$ is the *a priori* confidence in the hypothesis.

These 3 scores in the product of MAP score are defined in more detail below.

7.2.4.1 Kinematics Association Scoring

The association hypothesis kinematics scoring for a new incoming sensor report, $Y(S)$ to an existing track, $Y(T)$ assumes a Gaussian distribution [ellipsoid], with a sensor track covariance P which models the error in the track location due to possible motion. The baseline uses a position-only 3×3 scoring due to the uncertainty-in-the-uncertainty in the track velocity error as in the existing SUT. Thus, the kinematics score is computed as follows:

$P(Y(S)|Y(T), H) = \{1/(2\pi)^{d/2} |V|^{1/2}\} \exp[-1/2(I^T V^{-1} I)]$
where

- $Y(S)$ are the sensor report Gaussian kinematics with covariance R which for (E,N,Up) position only is a 3×3 ,
- $Y(T)$ are the track Gaussian kinematics with covariance P_k^+ ,
- H on the left side of the equation is the hypothesis that the report and track are associated,
- d is the dimension of the Gaussian kinematics state,
- $|V|$ is the determinant of the innovations covariance, $V = H[P_k^+ H^T + R]$. H is the measurement to state conversion matrix. For position only reports H is the 3×6 conversion matrix to 3-D from track 6-D with identity in the first 3×3 and remainder zeros.
- I is the innovations vector, $I = Y(S) - H Y(T)$

7.2.4.2 Noncommensurate Attributes (Entity Type) Association Scoring

The measured sensor attributes are typically commensurate (e.g., for ESM RF, PRI, scan rate, PW) and can be included using commensurate scoring which is similar to the kinematics scoring above, for the Gaussian error models. The use of the reported entity ID confidence vector helps to track through crossing entities of different types. However, the errors in these reported entity ID confidence vectors are highly correlated with unknown correlations. The default entity attribute scoring assumes that these errors are uncorrelated given the entity class (i.e., noncommensurate ID confidence vector) to aid in report-to-track association for differing entity types. The fact that this is not the case for one sensor will cause the expected value for this term to not be 1 and will thus increase the association score over the non-association score. This penalty is not severe here due to the inaccuracies in the *a priori* P_D and P_{FA} sensor information. Even though noncommensurate scoring is assumed to be sufficient, noncommensurate ID updating is not used for the ID confidence vector output to Tier 1 fusion. The score for noncommensurate entity type report inputs is as follows:

$$P(Z(S), Z(T)|Y(S), Y(T), H) = \{\sum_k [P(K|Z(T), Y(T), H)] P(K|Z(S), Y(S), H) / P(K|Y(T), Y(S), H)\}$$

where

- $Y(S)$ is the sensor report Gaussian kinematics with covariance R ,
- $Y(T)$ is the track Gaussian kinematics with covariance P_k^+ ,
- H is the hypothesis that the report and track are associated,
- K are the elements of the disjoint ID classes (e.g., friend, foe, neutral)
- $Z(T)$ are the parameters and attributes for the reports associated with the track,
- $Z(S)$ are the parameters and attributes from the sensor report,
- $P(K|Z(T), Y(T), H)$ are the elements of the sensor track classification confidence vector,
- $P(K|Z(S), Y(S), H)$ are the elements of the sensor report classification confidence vector, and
- $P(K|Y(T), Y(S), H)$ are the elements of the *a priori* (unclassified) confidence vector which is a user input with default to uniform (i.e., all .2 values having 5 classes).

7.2.4.3 A priori Association Hypothesis Scoring

For the *a priori* hypotheses terms, $P(H)$, the following is the 0th-order approximate scoring equation for each sensor report S and track T hypothesis as in the existing SUT, viz.,

$$\begin{aligned} P(\text{association}) &= [1 - P_{FA}(S)][1 - P_{FA}(T)]P_D(S)P_D(T) \\ P(\text{pop-up}) &= [1 - P_{FA}(S)][1 - P_D(T)]P_D(S) \\ P(\text{false alarm}) &= P_{FA}(S)P_D(S) \\ P(\text{propagate}) &= [1 - P_{FA}(T)][1 - P_D(S)]P_D(T) \\ P(\text{drop}) &= P_{FA}(T)P_D(T) \end{aligned}$$

where

- $P_D(S)$ is the probability of detection given in the sensor, S , report (i.e., an approximation to the probability that the sensor detects a current sensor track),
- $P_{FA}(S)$ is the probability of this sensor report being false alarm (FA) as given in the sensor report (e.g., expected number of false reports divided by total expected number of reports),
- $P_D(T)$ is the probability that the sensor file will have the reported entity as a track (i.e., probability sensor will have had a prior track initiation of this object). The default is the $P_D(S)$ for the last associated report.
- $P_{FA}(T)$ is the probability that this track is a false alarm (e.g., expected number of false tracks divided by total expected number of tracks). The is the $P_{FA}(S)$ for the last associated report.

7.2.4.4 Hypothesis Evaluation Summary

The total scene hypothesis score is the product of the individual hypothesis scores for how all the given batch of reports and the sensor tracks are associated (i.e., for

each of the 5 types of hypotheses). These five association type scores for each report or track using the sensor classification and *a priori* confidences are as follows:

1. Association Hypotheses

$$P(Y(S)|Y(T),H)P(Z(S), Z(T)|Y(S), Y(T), H)P(H) = \{|V|^{-1/2}\} \exp[-\{I^T V^{-1} I\}/2] \{\sum_k [P(K|Z(T), Y(T), H) P(K|Z(S), Y(S), H)/P(K|Y(T), Y(S), H)]\} [1 - P_{FA}(S)] [1 - P_{FA}(T)] P_D(S) P_D(T)$$
2. Pop-up Hypotheses

$$P(Y(S)|H)P(Z(S)|Y(S),H)P(H) = E(|V|)^{-1/2} \exp[-\mu/2] [1 - P_{FA}(S)] [1 - P_D(T)] P_D(S)$$
3. False Alarm Hypotheses

$$P(Y(S)|H)P(Z(S)|Y(S),H)P(H) = E(|V|)^{-1/2} \exp[-\mu/2] P_{FA}(S) P_D(S)$$
4. Propagation Hypotheses

$$P(H) = [1 - P_{FA}(T)] [1 - P_D(S)] P_D(T)$$
5. Track Drop Hypotheses

$$P(H) = P_{FA}(T) P_D(T)$$

where

- $Y(S)$ are the sensor report Gaussian kinematics with covariance R ,
- $Y(T)$ are the track Gaussian kinematics with covariance P_k^+ ,
- H in the overall equations is the hypothesis that the report and track are associated,
- $|V|$ is the determinant of the innovations covariance,

$$V = H [P_k^+] H^T + R,$$
- H in the innovations covariance equation is the measurement to state conversion matrix,
- E is the expectation operation,
- I is the innovations vector,

$$I = Y(S) - H Y(T)$$
- K are the elements of the disjoint class network,
- $Z(T)$ are the parameters and attributes from the track,
- $Z(S)$ are the parameters and attributes from the sensor report,
- $P_D(S)$ is the probability of detection by the sensor, S , of the hypothesized associated object,
- $P_{FA}(S)$ is the probability of false alarm (FA) of the sensor for this type of report,
- $P_D(T)$ is the probability of detection of this object in the sensor track file,
- $P_{FA}(T)$ is the probability that this track is a false alarm.

For the non-association report hypotheses (i.e., pop-up initiation, and false alarm) the expected value of the kinematics score is used as in the existing SUT. Namely, the kinematics score equation is used except that the chi-square statistic (i.e., $[I^T V^{-1} I]$) is replaced with its mean, μ . Namely,

1. $\mu = 0.455$ for 1 degree of freedom (DOF) (e.g., bearings-only)
2. $\mu = 1.39$ for 2 DOF (e.g., x and y only)
3. $\mu = 2.37$ for 3 DOF (e.g., Cartesian (x, y, z))
4. $\mu = 3.36$ for 4 DOF (e.g., 2 dimensions with rates)

5. $\mu = 4.35$ for 5 DOF

6. $\mu = 5.35$ for 6 DOF (e.g., Cartesian (x, y, z) with rates)

The expected innovations covariance multiplier is the expected value of the report to track innovations covariance in the determinant term of the kinematics score. Namely, it is the expected value of $|V|^{-1/2}$ for the given report as in the existing SUT. For the non-association track hypotheses (i.e., propagation, and drop track), the kinematics, $P(Y(T))$, and noncommensurate terms are all constant with respect to the maximization so are ignored as in the existing SUT.

7.2.5 Hypothesis Selection

The objective of this function is to select the association (and non-association) hypotheses that are used for state estimation based upon hypothesis evaluations. The Vogel, best association score selected first, search algorithm is used.

For report-to-track hypothesis selection, all unassociated reports initialize a new track and a sensor track is dropped after it has not been updated for a sufficient period of time. The sensor hypothesis selection function declares a sensor track to be deleted based upon the elapse of time without an association when one is expected. The sensor tracks are maintained until they are unassociated for longer than the user-specified length of time (e.g., 20 seconds) as in the existing SUT. When a sensor track is dropped, a delete track number message is sent to the ownship fusion node.

7.2.6 State Estimation

The primary sensor state estimation design operations are the following:

1. Use the report and the associated last updated sensor track, propagated to the current report time, to update the sensor track kinematics
2. Use the new sensor report classification confidences to replace the last ID confidence vector.
3. Update the P_D and P_{FT} for each track using the P_D and P_{FA} for the associated report.

7.2.6.1 Kinematics State Estimation

Given a sensor report and its associated sensor track, the track kinematics state and its covariance is updated using a Kalman filter as in the existing SUT. One difference is that the radar report is a 5 vector with a 5×5 error covariance, R , used in the Kalman filter update instead of a 3×3 . Namely, for a 3-D report update

$$Y_k(T) = Y_k^+(T) + K[Y_k(S) - H Y_k^+(T)]$$

$$P_k = [I - KH] P_k^+$$

where

- $Y_k(T)$ is the updated sensor track T state at time increment k .

- $Y_k(S)$ are the sensor report, S , Gaussian kinematics with 3×3 error covariance, R , at time increment k .
- K is the 6×3 Kalman gain matrix,

$$K = P_k^+ H^T [H P_k^+ H^T + R]^{-1}$$
- I is the identity matrix.
- $Y_{k-1}(T)$ is the best estimate at time $k-1$ of the sensor track T kinematics state which consists of position and velocity in x and y .
- $Y_k^+(T)$ is the propagated estimate of the sensor track T kinematics state and before an update at time k .
- P_k is the error covariance for Y_k at time k . The functional sensor T dependence is suppressed since it would appear for all covariances P of ownship tracks, T .
- P_k^+ is the error covariance for Y_k^+ at time k before an update.
- H is the measurement to state conversion matrix that is the 3×6 conversion matrix to 3-D from track 6-D with identity in the first 3×3 and remainder zeros.

The positive definiteness of the covariance is essential for the filter. To avoid such a problem with minimum computational cost, it is recommended that the matrices be made symmetric by placing the lower left part of P into the upper right each update (i.e., since the lower left is less sensitive to round-off errors). For the ownship tracks that are not updated, the propagated state, $Y_k^+(T)$ and its covariance, P_k^+ , are used as the current state, $Y_k(T)$, and its covariance, P_k . The initial state error covariance is the report error covariance R in the 3×3 position and a parameter specified error covariance in velocity based upon entity type velocity uncertainties. The baseline is 300 m/sec one sigma in E and N and 50 m/sec when Up is added. The current report and the updated track state are included in the sensor track file to be passed to the ownship fusion node.

7.2.6.2 Entity Type State Estimation

The sensor report classification confidence vector is used to replace the last ID confidence vector.

7.2.6.3 Sensor Track Confidence Estimation

The current associated report P_D and P_{FA} (i.e., as used in the hypothesis evaluation) is attached to each updated track as the track P_D and P_{FT} . The propagated tracks retain their P_D and P_{FT} .

7.3 Tier 1 Ownship Fusion Node Detailed Design

7.3.1 Data Preparation

The ownship tracks are propagated to the current sensor track file time. All the current sensor and ownship tracks have a common time. This is done for the ownship states and their covariance matrices via multiplication by a time dependent ϕ matrix and the addition of noise for

the uncertainty in the entity dynamics over this time period.

7.3.2 Data Association

The primary ownship to sensor data association design operations are the following:

1. **Hypothesis Generation:** Determine which confirmed track-to-track associations are within confirmed track gates, then remove from both lists and pass to hypothesis selection. For the remainder find feasibly associated ownship tracks within gates of each sensor track using the most current ownship tracks propagated to the sensor track time.
2. **Hypothesis Evaluation:** Compute MAP scores for all feasible (unconfirmed) track-to-track association hypotheses.
3. **Hypothesis Selection:** Associate confirmed associations that are within confirmed track gates. Apply assignment algorithm to find best track-to-track associations and make ownship track initiation and deletion. Make confirmed association decisions.

7.3.3 Hypothesis Generation

7.3.3.1 Confirmation Gating

The current sensor track is gated with the ownship track to which the sensor track number has been confirmed to insure that the ownship track has not been pulled off by other source reports/tracks since the last sensor update. The baseline design uses the 3-D position-only chi-square gate to avoid detrimental affects of inaccurate velocity estimates especially over long revisit times or uncertain ID estimates. The kinematics gate is user defined (e.g., $M=49$ for a 7 sigma gate). Namely, for the sensor track position-only, $Y(S)$, gate out all ownship tracks, $Y(T)$, such that

$$I^T V^{-1} I > M$$

where

- I is the 3×1 innovations column vector,

$$I = Y(S) - H Y_k^+(T)$$
- H is the measurement to state conversion matrix that is the 3×6 conversion matrix to 3-D from track 6-D with identity in the first 3×3 and remainder zeros.
- V is the innovations covariance,

$$V = H [P_k^+] H^T + R$$
- $Y(S)$ is the 3×1 sensor report Gaussian kinematics with 3×3 error covariance R , which is the covariance of the measurement error in E , N , Up coordinates
- $Y(T)$ is the 6×1 track Gaussian kinematics with covariance P_k^+ propagated to current time above in data preparation.
- M is the gate around $Y(S)$, outside of which $Y(T)$ is no longer confirmed, for the baseline a 7-sigma gate is used.

2-D-only position gating is used. If all gates are passed then the pair is marked as the only feasible associations for each other (i.e., removed for consideration for further association). This presumes that each sensor and ownship track is a track on a single entity. If the gates are not passed, the confirmation is removed, and the pair is passed along for data association and specifically passed next to the rest of hypothesis generation for feasible association gating.

7.3.3.2 Gating for New Sensor Tracks

The next step is to find remaining unconfirmed ownship tracks (i.e., ownship tracks whose association is not confirmed with any ownship track) within 3-D gates of each unconfirmed sensor track. 2-D gates are used. Gating is performed for each unconfirmed sensor track against all unconfirmed ownship tracks. Each ownship track passing the gates is included in the feasibly associated tracks for each unconfirmed sensor input.

7.3.4 Hypothesis Evaluation

The baseline scoring equations with entity type classification network confidences are based upon the standard Max a Posteriori (MAP) hypothesis evaluation scoring defined as follows:

$$\begin{aligned} \max P(H|R) &= \max \{P(R|H) P(H)\} \\ &= \max \{P(Y|H) P(Z|Y, H) P(H)\} \\ &= \max[\prod_T \{P(Y(S)|Y(T), H) P(Z(S), \\ &\quad Z(T)|Y(S), Y(T), H) P(H)\}] \end{aligned}$$

where

- The maximization's are over all association and non-association hypotheses, H ,
- H is the set of feasible association or non-association hypotheses,
- R represents both the ownship track, T , and sensor track, S , data,
- Y is the set of kinematics from both,
- Z is the set of all parameters & entity attributes measured in both
- the product is over all independent track, T , hypotheses (i.e., of all 5 types),
- $Y(T)$ are the track kinematics, the $P(Y(T)|H)$ term is dropped as constant with respect to the maximization,
- $Y(S)$ are the sensor track kinematics,
- $Z(T)$ are the parameters and entity attributes from the ownship track,
- $Z(S)$ are the parameters and entity attributes from the sensor track,
- $P(H)$ is the *a priori* confidence in the hypothesis.

These 3 scores in the product of MAP score are defined in more detail below.

7.3.4.1 Kinematics Association Scoring

The association hypothesis kinematics scoring for a new incoming sensor track, $Y(S)$ to an existing ownship track, $Y(T)$ assumes a Gaussian distribution (ellipsoid), with a ownship track covariance P which models the error in the track location due to possible motion. Then the kinematics score is computed as follows:

$$P(Y(S)|Y(T), H) = \{1 / [(2\pi)^{d/2} |V|^{1/2}]\} \exp[-1/2 \{I^T V^{-1} I\}]$$

where

- $Y(S)$ are the sensor track Gaussian position kinematics with covariance R ,
- $Y(T)$ are the ownship track Gaussian kinematics with covariance P_k^+ ,
- H on the left side of the equation is the hypothesis that the sensor track and ownship track are associated,
- d is the dimension of the Gaussian kinematics state which here is a constant = 3 since only position is used for association hypothesis evaluation,
- $|V|$ is the determinant of the innovations covariance, $V = H [P_k^+] H^T + R$. H is the sensor track position to ownship state conversion matrix that is the 3x6 conversion matrix to 3-D from track 6-D with identity in the first 3x3 and remainder zeros. Note that H is 3x3 identity when P is the position-only covariance of the track state in xyz.
- I is the innovations vector,
 $I = Y(S) - H Y(T)$

7.3.4.2 Noncommensurate Attributes (Entity Type) Association Scoring

The noncommensurate ID confidence vector scoring used in the Tier 0 fusion nodes are used here except that the pedigree of the ownship track ID confidence is maintained. The current sensor ID confidence vector is combined with the noncommensurate portion of the track ID confidence vector. For example, when radar is being fused with the ownship track file containing ESM and IRST ID contributions, then the ESM and IRST updated track ID confidence vector maintained in the track ID pedigree is used for the track ID confidence vector instead of the all sensor ownship track ID confidence vector. The score for noncommensurate entity type track inputs is the same as above, viz.,

$$P(Z(S), Z(T)|Y(S), Y(T), H) = \{\sum_K [P(K|Z(T), Y(T), H) P(K|Z(S), Y(S), H) / P(K|Y(T), Y(S), H)]\}$$

where only terms for which the *a priori* (i.e., $P(\text{unclassified})$) $P(K|Y(T), Y(S), H) \neq 0$ are used

- $Y(S)$ are the sensor track Gaussian kinematics with covariance R ,
- $Y(T)$ are the ownship track Gaussian kinematics with covariance P_k^+ ,
- H is the hypothesis that the sensor track and ownship track are associated,
- K are the elements of the disjoint class network,

- $Z(T)$ are the parameters and attributes for the sensor track associated with the ownship track,
- $Z(S)$ are the parameters and attributes from the sensor track,
- $P(K|Z(T), Y(T), H)$ are the elements of the noncommensurate portion of the ownship track classification confidence vector,
- $P(K|Y(T), Y(S), H)$ are the elements of the *a priori* (unclassified) confidence vector,
- $P(K|Z(S), Y(S), H)$ are the elements of the sensor track classification confidence vector.

7.3.4.3 *A priori* Association Hypothesis Scoring

For the *a priori* hypotheses terms, $P(H)$, the same 0th-order approximate scoring equation for each sensor track S and ownship track T hypothesis is used, viz.,

$$\begin{aligned} P(\text{association}) &= [1 - P_{FA}(S)][1 - P_{FA}(T)] P_D(S) P_D(T) \\ P(\text{pop-up}) &= [1 - P_{FA}(S)][1 - P_D(T)] P_D(S) \\ P(\text{false alarm}) &= P_{FA}(S) P_D(S) \\ P(\text{propagate}) &= [1 - P_{FA}(T)][1 - P_D(S)] P_D(T) \\ P(\text{drop}) &= P_{FA}(T) P_D(T) \end{aligned}$$

where

- $P_D(S)$ is the sensor track probability of detection passed by the sensor fusion node. This term is the probability of detection for the report last associated with the track.
- $P_{FA}(S)$ is the probability of this sensor track being false (e.g., expected number of false tracks divided by total expected number of tracks). This term is approximated by the track P_{FT} passed from the sensor fusion node which is the P_{FA} of the last associated report.
- $P_D(T)$ is the probability that the ownship file contains a track that represents the sensor track. This is approximated by the last updated ownship track P_D .
- $P_{FA}(T)$ is the probability that this ownship track is a false alarm (e.g., expected number of false ownship tracks divided by total expected number of ownship tracks). This is approximated by the last updated ownship track P_{FT} .

7.3.4.4 Hypothesis Evaluation Summary

The total scene hypothesis score is the product of the individual hypothesis scores for how all the given batch of reports and the ownship tracks are associated (i.e., for each of the 5 types of hypotheses). These five association type scores for each sensor track and/or ownship track using the track kinematics, ID confidences, and *a priori* confidences are as follows:

1. Association (i.e., sensor track and ownship track combine) Hypotheses

$$P(Y(S)|Y(T), H) P(Z(S), Z(T)|Y(S), Y(T), H) P(H) = \{|V|^{-1/2}\} \exp[-1/2 \{I^T V^{-1} I\}] \{\Sigma_K [P(K|Z(T), Y(T), H)]$$

$$P(K|Z(S), Y(S), H) / P(K|Y(T), Y(S), H)\} [1 - P_{FA}(S)] [1 - P_{FA}(T)] P_D(S) P_D(T)$$

2. Pop-up Hypotheses
 $P(Y(S)|H) P(Z(S)|Y(S), H) P(H) = E(|V|)^{-1/2} \exp[-\mu/2] [1 - P_{FA}(S)] [1 - P_D(T)] P_D(S)$
3. False Alarm Hypotheses
 $P(Y(S)|H) P(Z(S)|Y(S), H) P(H) = E(|V|)^{-1/2} \exp[-\mu/2] P_{FA}(S) P_D(S)$
4. Propagation Hypotheses
 $P(H) = [1 - P_{FA}(T)] [1 - P_D(S)] P_D(T)$
5. Track Drop (i.e., delete ownship track) Hypotheses
 $P(H) = P_{FA}(T) P_D(T)$

where

- $Y(S)$ are the sensor track Gaussian kinematics with covariance R ,
- $Y(T)$ are the ownship track Gaussian kinematics with covariance P_k^+ ,
- H in the overall equations is the hypothesis that the report and track are associated,
- $|V|$ is the determinant of the innovations covariance,
 $V = H [P_k^+] H^T + R$,
- H in the innovations covariance equation is the measurement to state conversion matrix,
- E is the expectation operation,
- I is the innovations vector,
 $I = Y(S) - H Y(T)$
- K are the elements of the disjoint class network,
- $Z(T)$ are the parameters and attributes from the track (not used since pedigree used),
- $Z(S)$ are the parameters and attributes from the sensor track (not used since pedigree used),
- $P_D(S)$ is the probability of detection by the sensor, S , of the hypothesized associated object,
- $P_{FA}(S)$ is the probability of false alarm (FA) of the sensor for this track,
- $P_D(T)$ is the probability of detection of this object in the ownship track file,
- $P_{FA}(T)$ is the probability that this ownship track is a false alarm.
- $\mu = 2.37$ for the 3-D kinematics hypothesis evaluation

For the non-association sensor track hypotheses (i.e., ownship track initiation, and sensor false alarm) the expected value of the kinematics score is used. Namely, the kinematics score equation is used except that the chi-square statistic (i.e., $I^T V^{-1} I$) is replaced with its mean, μ , viz.,

1. $\mu = 0.455$ for 1 degree of freedom (DOF) (e.g., bearings-only)
2. $\mu = 1.39$ for 2 DOF (e.g., x and y)
3. $\mu = 2.37$ for 3 DOF (e.g., Cartesian (x, y, z))
4. $\mu = 3.36$ for 4 DOF (e.g., 2 dimensions with rates)
5. $\mu = 4.35$ for 5 DOF
6. $\mu = 5.35$ for 6 DOF (e.g., Cartesian [x, y, z] with rates)

The expected innovations covariance multiplier is the expected value of the sensor track to ownship track

innovations covariance in the determinant term of the kinematics score. Namely, it is the expected value of $|V|^{1/2}$ for the given sensor track. For the non-association ownship track hypotheses (i.e., ownship track propagation, and drop track), the kinematics, $P(Y(T))$, and noncommensurate terms are all constant with respect to the maximization, so are ignored.

7.3.5 Hypothesis Selection

The hypothesis selection objective and process is similar to Tier 0. For ownship track fusion there is an additional complication when a drop track message is received from the sensor. The ownship immediately drops the confirmed ownship track association, if any. When the sensor track is the only constituent part of the last associated ownship track, this solitary associated ownship track is dropped. When there is at least one other sensor constituent part, the track is retained but the current sensor constituent part flag is eliminated. The kinematics and ID pedigree is retained since the sensor track may have been valid and only dropped due to sensor mode or FOV changes due to aircraft maneuvering.

7.3.6 State Estimation

The primary ownship state estimation operations are the following:

1. Use the report for the newly updated sensor track (i.e., passed with the sensor track file) and the last updated ownship track (i.e., propagated to the current report time) to update the ownship track kinematics using a Kalman filter.
2. Use the sensor track classification confidences to update the noncommensurate portion of the ownship track ID pedigree.
3. Update the P_D and P_{FT} for each track using the P_D and P_{FA} for the associated report.

7.3.6.1 Kinematics State Estimation

Given the sensor reports associated with the sensor tracks, each ownship track kinematics state and its covariance can be updated using a Kalman filter such as used in Tier 0 fusion. For track initiation the sensor track state and covariance is used.

7.3.6.2 Entity Type State Estimation

The ownship track ID state is updated only when a new ID state is reported from a Tier 0 fusion node. The equation used to update an element C of the track type vector for each conditionally independent track ID confidence vector is as follows:

$$\text{if } P(C|Y(S), H) \neq 0 \\ P(\text{class } C|T, S, H) = \frac{[P(C|T, H)P(C|S, H)/P(C|Y(S), H)]}{\sum_k [P(K|T, H)P(K|S, H)/P(K|Y(S), H)]}$$

$$\sum_k [P(K|T, H)P(K|S, H)/P(K|Y(S), H)]$$

if $P(C|Y(S), H) = 0$

$$P(\text{class } C|T, S, H) = 0$$

where,

- C is the element of the ownship track class vector being updated,
- T is the ownship entity track data [both kinematics and attribute],
- S is the current sensor track data [both kinematics and attribute],
- $P(C|S, H)$ is the entity ID confidence vector from the current sensor fusion node,
- $P(C|T, H)$ is the ownship entity ID confidence vector containing only pedigree from noncommensurate sensors
- H is the association hypothesis where the associated sensor ID pedigree has been updated,
- K is the index of type classes [summed over for normalization],
- $P(C|Y(S), H)$ is the *a priori* probability of an entity of type C having the location and velocity given by the track kinematics state which is specified *a priori* in a table look-up, and each of the probabilities are the components of the noncommensurate report, track, and *a priori* entity type vectors

The most recent ID confidence vector from each associated noncommensurate sensor is used in the update. For example, for a radar-to-ownship entity confidence vector update where the ownship track already includes a full pedigree (i.e., radar, ESM, and IRST ID contributors), the above update is performed twice. First the ESM confidence vector updates the IRST ID confidence vector to form a new ownship entity ID confidence vector using the above equation. Second, the new radar entity ID confidence vector updates the resulting ownship ESM and IRST pedigree entity ID confidence vector using the above equation. The ownship track file is updated to contain this updated "best" track ID confidence vector as well as the "best" track ID confidence vectors from each noncommensurate sensor (i.e., radar, ESM, and IRST).

7.3.6.3 Ownship Track Confidence Estimation

The current associated track P_D and P_{FT} are compared with the ownship track P_D and P_{FT} . The higher P_D and lower P_{FT} are used. The propagated ownship tracks retain their P_D and P_{FT} .

7.4 Tier 2 Cooperative Fusion Node Detailed Design

For the baseline design each fighter shares its cooperative track file with the other fighters in its flight. These inputs to each fighter's cooperative fusion node are called the 'flight-cooperative tracks' which are fused

with the 'ownship-cooperative tracks' on each fighter in the Tier 2 cooperative fusion nodes. The Tier 1 'ownship tracks' are also be fused with the 'ownship-cooperative tracks' in the Tier 2 fusion nodes.

The sharing of cooperative track files provides the best information in each communications to lessen the impact of missed communications and potentially reduce the bandwidth and processing complexity and improve consistency. However it has the highest error correlations with the ownship track files and increases the impact of spoofing. The following sharing of cooperative track files is chosen as the baseline:

1. Report sharing to avoid cooperative track re-initialization to remove filtered track error autocorrelations over time at the cost of additional bandwidth
2. Ownship assigned track sharing with cooperative "track ownership" to avoid track error correlations by assigning the best source to provide updates for each track (used in similar sensor surveillance systems) in the cost of reduced synergy when each source provides different parts of the entity information (e.g., range, angle, resolution, IFF, type, ID).
3. Ownship track file sharing (e.g., with or without cooperative track associations) to lessen track error correlations by sharing separately derived ownship track files each of which are used to reinitialize the cooperative track files on each platform.

7.4.1 Data Preparation

Data Preparation transforms the current batch of input ownship tracks and the current ownship-cooperative tracks in preparation for data association which generates, evaluates, and selects from the alternative associations between the track files. The primary data preparation design operations include preprocessing of inputs to put into the ownship format and propagation to the most recent time

7.4.2 Data Association

The primary ownship-to-cooperative track association design operations are the following:

1. **Hypothesis Generation:** Determine if confirmed track-to-track associations are within confirmed track gates. Find feasibly associated ownship tracks within gates of each cooperative track using nearest in time last updated cooperative propagated to each feasible sensor track time.
2. **Hypothesis Evaluation:** Associate confirmed associations that are still within confirmed track gates. Compute MAP scores for all feasible (unconfirmed) association hypotheses.
3. **Hypothesis Selection:** Apply assignment algorithm to find best associations and make

ownship track initiation and deletion decisions. Make confirmed association decisions.

7.4.2.1 Hypothesis Generation

This function is performed as described for Tier 1 except for ownship-cooperative confirmed track and unconfirmed track gating instead of sensor-ownship gating.

7.4.2.2 Hypothesis Generation

This function is performed as described for Tier 1 except for using ownship-cooperative tracks instead of using sensor-ownship tracks for unconfirmed association hypothesis evaluation using kinematics, ID, and *a priori* terms. Specifically, the noncommensurate sensor ID pedigree (i.e., radar, ESM, and IRST for air-to-air) for each cooperative track is maintained and shared.

7.4.2.3 Hypothesis Selection

This function is performed as described for Tier 1 except for using ownship-cooperative tracks instead of using sensor-ownship tracks for unconfirmed association hypothesis selection. Namely, the baseline uses Hungarian algorithm for an optimal 2-D assignment. Similar drop track logic is used. Namely, when a drop track number message is received from the ownship or flight-cooperative fusion node, the cooperative fusion node immediately drops the corresponding confirmed cooperative track association, if any. When the dropped track is the only constituent part of the last associated cooperative track, this solitary associated cooperative track is dropped. When there is at least one other cooperative track constituent part, the track is retained, but the current cooperative constituent part in the cooperative track contribution pedigree is eliminated. The kinematics and ID pedigree is retained since the cooperative track may have been valid and only dropped due to source tracking problems.

7.4.3 State Estimation

The primary cooperative track state estimation operations are the following:

1. The current ownship or cooperative track kinematics is used to update the associated ownship-cooperative track using a Covariance Intersection (CI) filter.
2. The current ownship or cooperative track classification confidences is used to update the noncommensurate portion of the associated ownship-cooperative track ID pedigree.
3. The P_D and P_{FT} for each ownship-cooperative track is updated using the P_D and P_{FA} for the associated ownship or cooperative track.

7.4.3.1 Kinematics State Estimation

Since only track data is available for kinematics state estimation a Kalman filter is not used and a Covariance Intersection (CI) filter is used. CI provides provably consistent estimates that are derived without independence assumptions.

7.4.3.2 Cooperative Track ID State Update

The cooperative track ID state is updated only when a new track ID state is input (i.e., updated ownship track ID confidence vector). The equation used to update an element C of the cooperative track ID confidence vector

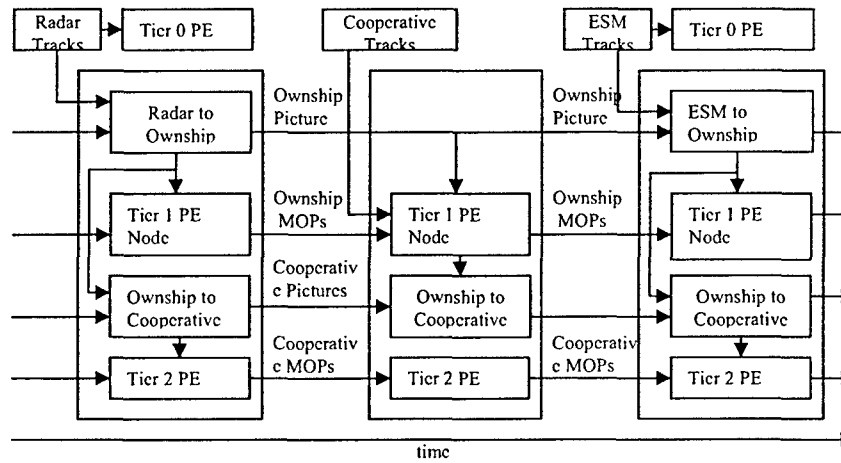


Figure 8: Ownship Fusion PE Network Nodes

with a noncommensurate ID pedigree confidence vector is same as that for Tier 1. The best noncommensurate ID confidence vector from each source is used for the update.

For example, for an ownship-cooperative entity confidence vector update where the ownship and the ownship-cooperative track both include a full pedigree (i.e., radar, ESM, and IRST ID contributors), the above update is performed twice. First the best ESM confidence vector updates the best IRST ID confidence vector using the noncommensurate update equation. Second, the best radar entity ID confidence vector updates the resulting track ID confidence vector. The best ID confidence vector from each sensor is the one with the highest single confidence component.

7.4.3.3 Cooperative Track Confidence Estimation

The current associated track P_D and P_{FT} are compared with the ownship-cooperative track P_D and P_{FT} . The higher P_D and lower P_{FT} are chosen. The propagated ownship tracks retain their P_D and P_{FT} .

7.5 Case Study Test Articles: Testbed Model and Implementation

7.5.1 Case Study PE Fusion Network

Each PE node associates the fused track files to truth to estimate the track file accuracy of Tier 0, 1 and 2 fusion nodes in each time period. In addition, PE associates the track files to each other to estimate the consistency of the ownship and cooperative fusion tracks over time. Use of truth for the associated report is not always viable as a poor accuracy report will not shift track state estimates sufficiently and truth is not available for AFFTC range

operations (i.e., truth is not available in the absolute sense). Figure 8 shows the ownship PE node network for the case study.

7.5.2 PE Assignment Algorithm

Based on the MAP score, the association matrix for hypothesis selection is generated. The MAP scores are normalized with $-\log$ and inserted into the assignment matrix. The Hungarian algorithm is used for optimizing the hypothesis selection. Table 1 shows the conversion of association matrix to 2-D assignment problem for Hungarian algorithm. "No Track Association" columns have been added to denote the hypothesis, H_2 , of a truth with no associated track. "False Track" rows have been added to denote the hypothesis, H_1 , of a false track for unassociated tracks. The zeros in lower right box discourage selection of non-association hypotheses.

| | Current Tracks | | | No Track Association | |
|---------------|---------------------------|---------------------------|---------------------------|----------------------|----------------|
| | $-\ln[P(R_1, T_1 H)P(H)]$ | $-\ln[P(R_2, T_1 H)P(H)]$ | $-\ln[P(R_3, T_1 H)P(H)]$ | $-\ln[P(H_2)]$ | $-\ln[P(H_1)]$ |
| Current Truth | $-\ln[P(R_1, T_1 H)P(H)]$ | $-\ln[P(R_2, T_1 H)P(H)]$ | $-\ln[P(R_3, T_1 H)P(H)]$ | $-\ln[P(H_2)]$ | $-\ln[P(H_1)]$ |
| False Track | $-\ln[P(H_1)]$ | $-\ln[P(H_1)]$ | $-\ln[P(H_1)]$ | 0 | 0 |
| | $-\ln[P(H_1)]$ | $-\ln[P(H_1)]$ | $-\ln[P(H_1)]$ | 0 | 0 |
| | $-\ln[P(H_1)]$ | $-\ln[P(H_1)]$ | $-\ln[P(H_1)]$ | 0 | 0 |

Table 1: Association Matrix for 2D Assignment Problem.

7.5.3 Case Study Environment for SUT

The simulator has been developed in MATLAB 6.5. It incorporates targets, platforms, on-board sensors, filters, and run parameters. The current case study settings were as follows:

1. **Targets:** 6 targets
2. **Platforms:** 2 Platforms
3. **Sensors:** Each platform has 3 on-board sensors (Radar, ESM andIRST)
4. **Scenario:**
 - (i) Air-to-Air ofoffensive sweep of 2 platforms vs. 6 targets engaging simultaneously in pairs from left and right 45 degrees and center at the same ranges.
 - (ii) Sensor models have search and track modes with separate range limits, probability of detection (Pd), probability of false alarm/tracks (Pfa/Pft), measurement accuracies, resolution limits, and ID uncertainties

5. Internetted Fusion Node Tier 0, 1 and 2 Test Articles:

- (i) **Data Preparation:** Spherical co-ordinates [R,Az] converted to Cartesian co-ordinates [x, y].
- (ii) **Data Association:** Feasible gating, Max a Posteriori (MAP) scoring, and Vogel approximation based hypothesis selection with 5 consecutive missing scans to drop tracks.
- (iii) **State Estimation:** Extended Kalman filter kinematics with P_D, P_{FT} and ID confidence update.

| OSS | Range (min-max) | Az FOV | Elev FOV | Report Rate | Range R Rate Accuracy | Az/EI Accuracy | R RR Resolution | Angular Resolution | Report ID Range | Report ID Confidence |
|--------------|-----------------------|-----------|----------|-------------|-------------------------|----------------|-------------------|--------------------|-----------------|----------------------|
| Radar Search | 0.1 – 80 Km | ± 60 deg | ± 30 deg | 4 sec | 25 m 1 m/sec | 8/16 m rad | 100 m 5 m/scc | 50 m rad | N/A | N/A |
| Radar Track | 0.1 – 95 Km | ± 40 deg | ± 10 deg | 0.1 – 6 scc | 25 m 1 m/sec | 8/16 m rad | 100 m 5 m/scc | 50 m rad | 35 Km | 75 – 85% air |
| ESM Search | 0.1 – 100 Km | ± 120 deg | ± 60 deg | 2.5 scc | Passive 50% of Range | 24/48 m rad | 0 m | 0 m rad | 100 Km | 80 – 90% |
| ESM Track | 0.1 – 100 Km | ± 80 deg | ± 20 deg | 1 scc | Passive 30% of Range | 24/48 m rad | 0 m | 0 m rad | 100 Km | 90 – 95% |
| IRST Search | 0.1 – 80 Km > 20 K ft | ± 50 deg | ± 16 deg | 4 sec | Passive 40% of Range | 2/4 m rad | N/A | 4/8 m rad | 20 Km | 75 – 85% |
| IRST Track | 0.1 – 80 Km > 20 K ft | ± 5 deg | ± 4 deg | 0.2 sec | Passive 20% of Range | 1/2 m rad | N/A | 4/8 m rad | 40 Km | 80 – 90% |

Table 2: Air-to-Air On-Board Sensor Top-Level Model Parameters.

6. **PE Process:** 7 PE Node Networks
 - (i) 3 individual sensor, 2 own-ship, 1 distributed fusion track-to-truth PE nodes.
 - (ii) 1 internetted platforms track-to-track PE node network.
7. **Air-to-Air On-Board Sensor Model Parameters:**

- (i) **Probability of Detection:** 0.7, 0.4, and 0.8 for radar, ESM andIRST respectively, at reference range (i.e., min range+0.75 (max – min range)).
- (ii) **False Alarm Rate:** 10 per hour (constant).
- (iii) **Other Parameters:** See Table2.

8. **Sensor ID:** Sensor ID declarations were based upon entity class confusion matrix per sensor mode. The entity ID declaration and confidence vector is the output for each mode change. The approximation to *a posteriori* entity class vectors from the confusion matrix of entity class declarations is given as:

$$P(C|S) = \frac{P(D|C)P(C)}{\sum_K [P(D_K|C)P(C)]}$$

where *C* is the entity class and *P(D|C)* are the elements in the confusion matrix column (*K*) for the given entity class declaration. *P(C)* is the *a priori* entity class confidence before any measurements. Table 3 shows a sample ID classification for ESM.

| True Entity Type | P(Foe Fighter Declaration) | P(Foe Non-Fighter Declaration) | P(Friendly Fighter Declaration) | P(Friendly Non-Fighter Declaration) | Neutral Declaration | Unknown Declaration |
|----------------------|----------------------------|--------------------------------|---------------------------------|-------------------------------------|---------------------|---------------------|
| Foe Fighter | 0.80 | 0.05 | 0.03 | 0.01 | 0.01 | 0.10 |
| Foe Non-Fighter | 0.05 | 0.80 | 0.01 | 0.03 | 0.01 | 0.10 |
| Friendly Fighter | 0.03 | 0.01 | 0.80 | 0.05 | 0.01 | 0.10 |
| Friendly Non-Fighter | 0.01 | 0.03 | 0.05 | 0.80 | 0.01 | 0.10 |
| Neutral | 0.05 | 0.05 | 0.05 | 0.05 | 0.70 | 0.10 |

Table 3: Sample Sensor ID Classification.

8 Experimental Results and Analysis

8.1 Baseline Scenario:

The baseline 2 vs. 6 offensive sweep scenario has 6 foe fighters (targets) engaging simultaneously in pairs from left and right 45 degrees and center to achieve a simultaneous missile launch against the blues (platforms). The blue 1 launches AMRAAM missiles on reds (fighters) 1, 2 and 3, 4 respectively. The blue 1 launches AMRAAM for the second time against the

surviving red. Then the other blue turns towards reds 5, 6 and launches missile. All the red fighters are in a pair staggered formation with the trailing fighter off to the left or to the right, sufficient to be not resolvable by blue radar until after the final red turn.

8.2 Tier 0

We ran the simulation for Tier 0 from time periods 1 to 329 with an interval of 1 time period. The time period was 1 second. Figures 9a, 9b and 9c show the typical simulation output for Radar, ESM and IRST respectively.

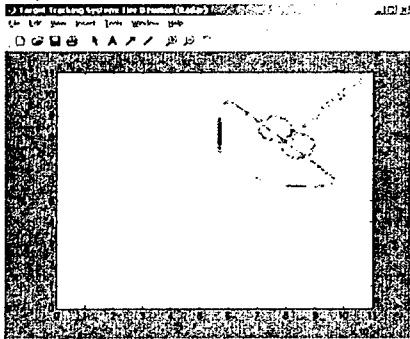


Figure 9a: Tier 0 SUT output for Radar

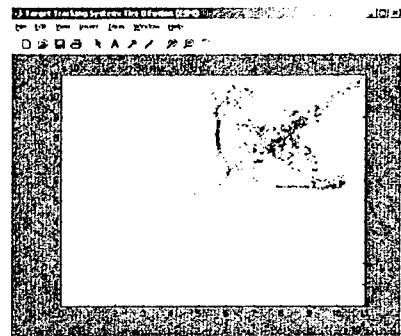


Figure 9b: Tier 0 SUT output for ESM.

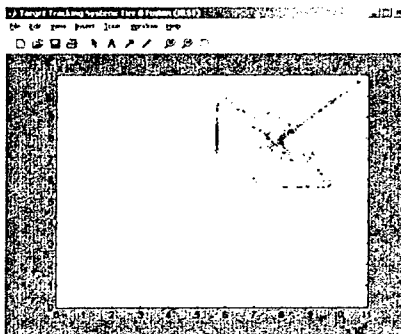


Figure 9c: Tier 0 SUT output for IRST.

The blue cross trajectory is for Platform 1 and the pink one is for Platform 2. The baseline 2vs6 offensive sweep scenario has 6 foe fighters coming towards 2 blue fighters with the objective of engaging at 10-15 km simultaneously in pairs from ± 45 degrees and center. The blue launch AMRAAM missiles between 20-25 km on 1,2 and 3,4, respectively. The second launch by blue 1 against the surviving red 3 occurs at about 10-15 km. Then the other blue turns at 5g towards red 5,6 and launches on 5,6. All fighters are in a pair staggered formation with the trailing fighter off to right and behind sufficient to be not resolvable by blue radar until after the final red turn.

The blue and red fighters are both initially in search mode for each other. Once the reds detect they turn off emissions and execute their pre-planned maneuvers to achieve near simultaneous launch on the projected blues. The reds all turn on their radars to lock-on to blues just after their last turn towards the projected blue position. The reds launch radar guided missiles at their closest blue targets as soon as possible. Red 5/6 should pull delaying turns together then turn towards an intercept with US 1 (i.e., highest closure rate) once their radar acquires rather than as shown in the Figure 9).

The blues split and turn towards the outside threats to take advantage of their longer range AMRAAM shots at each of outside red pairs. They support their launches until both outside reds are killed or until second shots are needed. In the baseline scenario shown, US2 achieves 2 kills with its first launches then turns towards reds 5/6 that have engaged US1 while taking its second shot at the surviving red 4. US1 will leave this second AMRAAM once it has acquired red 4, then pulls defensive maneuvers and countermeasures against the reds 5/6 missile launches while US2 completes red 5/6 kills.

The SUT gate multiplication factor was 5 and 15. The PE gate multiplication factors of 3 and 5, PE designs for Vogel and Hungarian based association, expected probability of false tracks, expected probability of detection and confidence ID updates.

8.3 Tier 1

Similar to Tier 0, the simulation for Tier 1 was run from time periods 1 to 329 with an interval of 1 time period. Figure 10 shows the simulation output for Platform 1 (blue cross) and Platform 2 (pink cross).

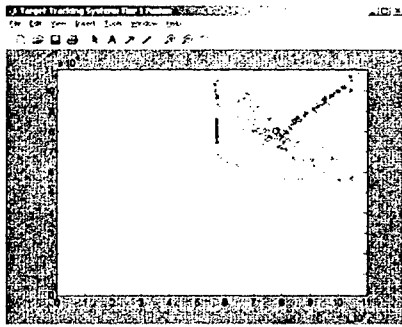


Figure 10: Tier 1 SUT output.

8.4 Tier 2

The simulation for Tier 2 was run from time periods 1 through 329 at an interval of 1 time period. Figure 11 shows the SUT simulation output for Tier 2.

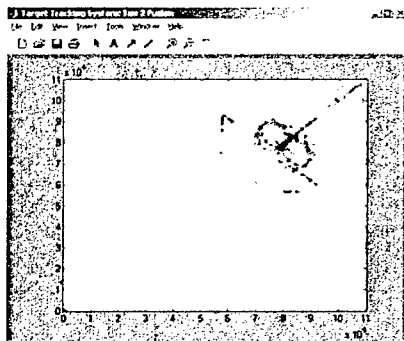


Figure 11: Tier 2 SUT output.

9 Design of Experiments

9.1 DOE Plan

We planned a Design of Experiments (DOE) scheme for the PE MoPs. We conducted these tests on Tier 0, Tier 1 and Tier 2. We decided on the following factors to setup the DOE:

- **Scenario Factors (Fixed):**
 - Offensive Sweep 2vs6 Air-to-Air
- **PE Factors:**
 - Design (Association)
 - Vogel Approximation (PE 1), and
 - Hungarian based association (PE 2)
 - Gating Factor: 3 and 5
- **System under test (SUT) Design Factors:**
 - Gating Factor: 5 and 15

So this yields a 2^k or 2^3 full factorial design. We used MINITAB to perform the DOE runs. The full factorial design details are as follows:

Factors: 3
Levels: 2

- (A) SUT Design Gating Factor
- (B) PE Gating Factor
- (C) PE Design

Base Design: 3, 8
Runs: 80
Replicates: 10
Blocks: 1
Center pts (total): 0

All terms are free from aliasing. The factors and interactions that are significant for various MoPs are denoted by 'S'. Tables 4, 5 and 6 show the summary of the DOE run results for Tier 0, 1 and 2 respectively.

In addition to these DOE runs, we ran another set of full factorial runs to see the effect of communication tiers on the various MoPs. We added another factor, (D) Tier, with two levels: Tier 1 and Tier 2. Table 7 shows the significant factors and their interactions for the various MoPs.

Tier 0:

| | | A | B | C | AB | AC | BC | ABC |
|------------------------|------------------------------------|---|---|---|----|----|----|-----|
| Track 1 to truth radar | Consistency | S | | | | | | |
| | Pft1 | S | S | | | | S | |
| | Mean location error | S | | | S | S | | |
| | Avg. std. deviation location error | S | | S | | S | | |
| | Avg. std. deviation | S | | S | S | | | |
| Track 2 to truth radar | Consistency | S | | S | | | | |
| | Pft | S | S | S | | | S | |
| | Mean location error | S | | | | | | |
| | Avg. std. deviation location error | S | | S | | S | | |
| | Avg. std. deviation | S | | S | | | | |
| Track 1 to truth ESM | Consistency | | | | | | | |
| | Pft | S | | | | | | S |
| | Mean location error | S | | | | | | |
| | Avg. std. deviation location error | S | | | S | | | |
| | Avg. std. deviation | S | | | S | S | | |
| Track 2 to truth ESM | Consistency | S | S | S | S | | | |
| | Pft1 | | | | | | | S |
| | Mean location error | S | | S | | | | S |
| | Avg. std. deviation location error | S | | | | | | S |
| | Avg. std. deviation | S | | | | | | |
| Track 1 to truth IRST | Consistency | S | | | | | | S |
| | Pft | S | | | | | | |
| | Mean location error | S | | | S | | S | |
| | Avg. std. deviation location error | S | | | | S | | |
| | Avg. std. deviation | S | | | S | S | S | |
| Track 2 to truth | Consistency | S | S | | | | | |
| | Pft | S | | | | | | |

| | | | | | | | | |
|-----------------------------|------------------------------------|---|---|---|---|---|---|---|
| truth IRST | Mean location error | S | S | | | | | S |
| | Avg. std. deviation location error | S | | | | | S | |
| | Avg. std. deviation | S | S | S | S | S | | S |

Table 4: Tier 0 DOE run summary.

10 Conclusions and Future Directions

10.1 Summary

In this and our prior works, we have described a formal and statistically-rigorous, fusion-based PE process for the evaluation of fusion-based tracking processes. In this current work we extended the prior accomplishments by adding significant improvements to the SUT to incorporate multiple and different on-board sensors along with 3 different tiers of data fusion and PE node management. Further we incorporated a systematic DOE implementation to rigorously analyze the different SUT and PE factors that would affect the MoPs. Among other improvements to the SUT, we incorporated asynchronous report and measurement generation along with more sophisticated data association algorithms for superior and consistent track pictures.

Tier 1:

| | | A | B | C | AB | AC | B C | ABC |
|-----------------------|-------------|---|---|---|----|----|--------|-----|
| Track to track | Consistency | S | S | S | S | | | |
| | Pft1 | S | S | S | S | | | |
| | Pft2 | | S | S | S | | | |

Inter Tier 1 and 2:

| | | A | B | C | D | AB | A C | AD | BC | BD | CD | AB C | ABD | AC D | BCD | ABC D |
|-------------------------|------------------------------------|---|---|---|---|----|--------|----|----|----|----|---------|-----|---------|-----|----------|
| Track to track | Consistency | | | | S | | | | | S | S | | | | | |
| | Pft1 | | | | S | | | | | S | S | | | | | |
| | Pft2 | S | S | | S | | | | | S | | S | | | | |
| | Mean location error | | | | S | | | S | | | | | | | | |
| | Avg. std. deviation location error | S | S | S | S | | | S | S | S | S | | | | | |
| | Avg. std. deviation | S | S | S | S | | | S | | S | S | | | | | |
| Track 1 to truth | Consistency | S | S | | S | | | S | | | | S | | | | |
| | Pft | S | | | S | | | S | | | | | | S | | S |
| | Mean location error | | | | S | | | | | | | S | | | | S |
| | Avg. std. deviation location error | | | | S | | | | | | | | | | | S |
| | Avg. std. deviation | S | S | S | S | | | S | | | S | | S | S | | S |
| Track 2 to truth | Consistency | S | | | S | | | S | | | | S | | | S | S |
| | Pft | | | | S | | | S | | | | S | | | S | S |
| | Mean location error | | | | S | | | | S | | | S | | | S | S |
| | Avg. std. deviation location error | S | S | | S | S | | | | | | | S | | S | S |
| | Avg. std. deviation | S | | S | S | | | S | S | | | S | | | S | S |

Table 7: Inter Tier 1 and 2 DOE run summary

| | | | | | | | | |
|-------------------------|------------------------------------|---|---|---|---|---|---|---|
| | Mean location error | S | S | S | S | | S | |
| | Avg. std. deviation location error | S | S | S | | | S | |
| | Avg. std. deviation | S | S | S | S | | S | |
| Track 1 to truth | Consistency | S | S | S | S | | | |
| | Pft | S | S | S | S | S | | |
| | Mean location error | S | | | S | | | |
| | Avg. std. deviation location error | | | S | S | | | |
| | Avg. std. deviation | S | S | S | S | S | S | S |
| Track 2 to truth | Consistency | S | S | S | S | S | | S |
| | Pft | S | S | | S | S | | S |
| | Mean location error | S | S | | S | | | |
| | Avg. std. deviation location error | S | S | | | | | |
| | Avg. std. deviation | S | S | S | S | S | S | S |

Table 5: Tier 1 DOE run summary.

Tier 2:

| | | A | B | C | A B | A C | B C | ABC |
|-----------------------|------------------------------------|---|---|---|--------|--------|--------|-----|
| Track to track | Consistency | | | | | | | S |
| | Pft1 | | | | | | | |
| | Pft2 | | | | | | | |
| | Mean location error | | | | | S | | |
| | Avg. std. deviation location error | | | | | | | |
| | Avg. std. deviation | | | | | S | | |

| | | | | | | | | |
|-----------------------------|---------------------------------------|--|--|--|--|--|---|---|
| Track 1 to truth | Consistency | | | | | | | |
| | Pft | | | | | | | |
| | Mean location error | | | | | | | |
| | Avg. std. deviation location error | | | | | | | |
| | Avg. std. deviation | | | | | | | |
| Track 2 to truth | Consistency | | | | | | S | |
| | Pft | | | | | | | S |
| | Mean location error | | | | | | S | |
| | Avg. std. deviation location error | | | | | | | |
| | Avg. std. deviation | | | | | | S | |

Table 6: Tier 2 DOE run summary.

The PE framework in our research enables reusable DF&RM avionics PE software developed specifically for AFFTC. This effort extended a previous AFOSR/AFFTC effort on PE methodology for AFFTC and implemented a PE network performing distributed fusion track-to-truth association for track file accuracy and distributed fusion track-to-track association for track file consistency. The distributed fusion PE software was also used (leveraged) for a project on warfare modeling by National Imagery and Mapping Agency (NIMA) at University at Buffalo.

10.2 Future Work

There are yet numerous other issues that can be addressed in this research, as PE is an under-researched topic in the Data Fusion community. For the focal problem types considered here, involving tactical aircraft applications, we see various possible future directions and enhancements, as described below.

10.2.1 Gating

In this paper, we used a 2-D-only position gating (i.e., not using altitude). We plan to incorporate user flags for max speed gates or strict type gating in the future. We presumed that each sensor and sensor track is a track on a single entity. Set covering approaches that deal with poor resolution issues are to be decided in the future.

10.2.2 Sensor attribute measurements

In this paper, the sensor attribute measurements are not modeled. The user needs to decide if he wants to have the reported entity confidence vector ID used in the report-to-track association hypothesis scoring. For this current work, the user flag was set to select entity typing confidences to be used for association scoring. In the future, when the entity type uncertainty-in-the-uncertainty is high this score will be avoided. Also when the entity type declaration has very high confidence (i.e., to the extent that it is always believed), then this scoring will be replaced with gating and replacement in state estimation. If this flag is not set the type confidences can

still be selected for use in state estimation for sensor track type updating to improve the sensor track type confidences. The expected value of these terms is one so not selecting entity type scoring will not affect the normalization of the kinematics and *a priori* non-association scores. The sensor report classification confidence vector is used to replace the last ID confidence vector. In the future we plan to consider the replacement only if the sensor has not changed to a worse ID mode (i.e., from track back to search).

10.2.3 Report-to-track

For report-to-track hypothesis selection, all unassociated reports initialize a new track and a sensor track is dropped after it has not been updated for a sufficient period of time. In future, this track drop logic depends upon whether the track is tentative or is declared to be of high confidence (i.e., validated) and the sensor mode.

10.2.4 Data Preparation

The primary data preparation design operations include preprocessing of inputs to put into the ownship format and propagation to the most recent time. In future, due to possible communications errors (e.g., modeled in future spirals) the data preprocessing functions may include the following:

1. Incomplete data: delete messages with data missing any vital field to include source, time, range, azimuth, and elevation.
2. Duplicate messages: delete messages which are exact duplicates.
3. Too old: delete sensor tracks that are older than a user specified parameter (e.g., 20 seconds).
4. Coordinate transformation: conversion into a common Cartesian track file coordinate system as necessary.
5. Propagation: propagation of the oldest track file to the most current track file, for example propagation of the current but delayed flight-cooperative track file to the ownship-cooperative track file common time.

The first 3 are straightforward to implement. The 4th is not needed in the present case since the cooperative track files are all in the same earth-fixed center Cartesian coordinates where rotation and curvature of the earth are ignored for this spiral. The 5th is done via the propagation equations since all cooperative tracks will be maintained at a common time in the present case. In future, if there is no common time for the ownship cooperative tracks, then the insertion of the propagation as part of the track unique hypothesis generation will be assessed to reduce the computational burden.

10.2.5 SUT Track Filters

More sophisticated track filters will be considered if track accuracy becomes an issue in the future. Examples include the following:

- The interacting multiple model (IMM) which adapts to unknown or changing target motion
 - considers a fixed set of target motion models (differ in noise & structure)
 - probabilistically combines estimates of individual filters matched to these models to determine the weight for each model.
- Particle filters which use sequential Monte Carlo generation of track state hypotheses to overcome nonlinear and non-Gaussian dynamics by adapting the necessary linear models based upon innovations accuracy
- Unscented Filter which approximates the error distribution using deterministic sigma points for cases with significant nonlinearities and/or nonlinear target dynamics.

Historically much effort has been put into track estimation technology, so the payoff for additional research is not expected as high as for less mature DF technology. Consequently this effort will apply simple existing filters sufficient to drive the development and test of the high potential capabilities (e.g., track confidence estimation, adjudication management, alternative Tier 2 data sharing strategies, and distributed level 2 and 3 fusion and sensor management).

10.2.6 Bandwidth Issues

In the future, the PE distribution nodes will consider *Bandwidth Utilization* as another MoP to assess the peak and average percentage of communications bandwidth load for distributed fusion and adjudication. In later efforts the present PE methodology and the software can be extended to assess (i) Autonomous Aircraft Adaptive Management, and (ii) Information Sharing Strategies (ISS). Both of the programs are in line with Wirfel's [13] long term plans. Autonomous aircraft adaptive management help aircraft share threat information to optimize the probability of mission success with in-flight re-planning capability. These are done in conjunction with the time of engagement on target, Emission Control (EMCON), and limited expendable countermeasures (CMs), ISSs, and perimeter limits of action. As a part of PE of ISS, the PE methodology can be applied to measure the relative ISS performance in the delay in achieving the weapon handoff kinematics and ID requirements in multi-aircraft Air-to-Air (A/A) and Air-to-Ground (A/G) scenarios.

Acknowledgement

This effort was motivated by the AFFTC at Edwards AFB and funded by AFOSR. We gratefully acknowledge

the technical guidance from staff at AFFTC, and programmatic guidance from staff at AFOSR.

References

- [1] Bowman, C. *Unifying Data Fusion and Resource Management Software Development Approaches Using the Dual Node Network Architecture*. To appear in *Unification of Fusion Theories*, edited by F. Smarandache, University of New Mexico. 2005.
- [2] Bowman, C. *The Dual Node Network Data Fusion and Resource Management Architecture*. AIAA Intelligent Systems Conference, Chicago, IL, USA. 2004.
- [3] Bowman, C. *MINT Fusion Course*. 2004.
- [4] Bowman, C. *The Data Fusion Tree Paradigm and Its Dual*. Proceedings of 7th National Symposium on Sensor Fusion: Invited paper, Sandia Labs, NM, USA. 1994.
- [5] Ghosh Dastidar, S., Sambhoos, K., Bowman, C., and Llinas, J. *Performance Evaluation Methods for Data-Fusion Capable Tactical Platforms*. International Conference on Information Fusion. Philadelphia, PA, USA. July 22-25, 2005.
- [6] Hoffman, J.R., and Mahler, R.P.S. *Multi-Target Miss Distance via Optimal Assignment*. IEEE Transactions on Systems, Man and Cybernetics: 34(3). pp. 327-336. 2004.
- [7] IEEE STD 610.12. *Standard Glossary of Software Engineering Technology*. IEEE, NJ. 1990
- [8] Linn, R.J., Hall, D.L., and Llinas, J. *A Survey of Multi-sensor Data Fusion Systems*. Proceedings of the SPIE. - The International Society for Optical Engineering SPI, Orlando, FL, USA. Vol. 1470, pp. 13-29. 1991.
- [9] Llinas, J, Talreja, D., and Bowman, C. *Design of a Performance Evaluation Methodology for Data Fusion-based Multiple Target Tracking Systems: Part 2: Analysis Methodologies*. NSSDF JHAPL. 2004.
- [10] Rawat, S, Llinas, J., and Bowman, C. *Design of a Performance Evaluation Methodology for Data Fusion-based Multiple Target Tracking Systems*. Proceedings of the SPIE. San Diego, CA, USA. Vol. 5099, pp. 139-151. April 23-25, 2003.
- [11] Roy, J., and Bossé, É. *Definition of Performance Evaluation Methodology for Sensor Data Fusion Systems*. Defence Research Establishment, Valcartier: DREV-R-9423. Québec, Canada. 1995.

[12] Steinberg, A., and Bowman, C. *Rethinking the JDL Model*. National Symposium on Sensor and Data Fusion. Laurel, MD, USA. June 7-11, 2004.

[13] Wirfel, J.A. *Electronic Warfare Topics Briefing*. 2000.

Appendix

A. Albersheim's Approximation

Albersheim's method is an approximation based upon the estimating the required signal to noise ratio (SNR) and then computing the SNR at the desired ranges. The required SNR is computed as follows:

$$\text{SNR} = A + .12AB + 1.7B$$

where

$$A = \ln \{ .62 / P_{fa} \}$$

$$B = \ln \{ P_d / [1 - P_d] \}$$

P_{fa} is the probability of false alarm (example value E-6)

P_d is the probability of detection

Given a probability of detection at the baseline range, a baseline SNR is derived. For example, for $P_{fa} = E-6$ and $P_d = .35$, $A = 13.3$, $B = -.62$, and $\text{SNR}_b = 13.3 - .99 - 1.05 = 11.2$. The SNR at a desired range, SNR_r , is

$$\text{SNR}_r = \text{SNR}_b \{ R_b / R_r \}^n$$

where

SNR_b is the baseline SNR

R_b is the baseline range

R_r is the desired range

$n=4$ for active sensors and $n=2$ for passive sensors

Once the SNR_r at the desired range is determined the probability of detection at the desired range can be generated.

B. MINITAB Response Optimizer

The MINITAB Response Optimizer provides an optimal solution for the input variable combinations and an optimization plot over all responses (MOPs). MINITAB's Response Optimizer helps to identify the combination of input variable settings that jointly optimize a single response or a set of responses. Joint optimization must satisfy the requirements for all the responses, i.e. MOPs, in the set, which is measured by the composite desirability. The overall desirability is a measure of how well the user has satisfied the combined goals for all the selected responses. Overall desirability has a range of zero to one. One represents the ideal case; zero indicates that one or more responses are outside their acceptable limits. Composite desirability is the weighted geometric mean of the individual desirabilities for the responses.

MINITAB calculates an optimal solution and draws a plot. It employs a reduced gradient algorithm with

multiple starting points that maximizes the composite desirability to determine the numerical optimal solution. The optimal solution serves as the starting point for the plot. This optimization plot allows the user to interactively change the input variable settings to perform sensitivity analyses and possibly improve the initial solution. As, the optimization plot is interactive, the user can adjust input variable settings on the plot to search for more desirable solutions.

The optimization is accomplished by:

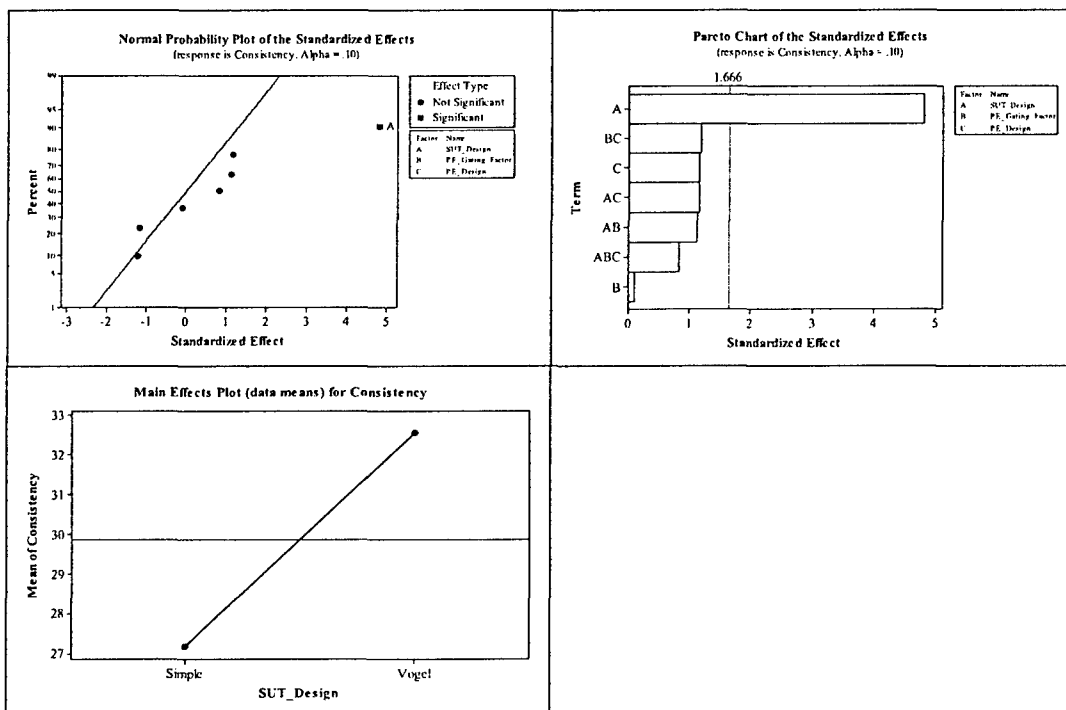
- Obtaining the individual desirability for each response.
- Combining the individual desirability's to obtain the combined or composite desirability.

C. Tier 0 DOE Charts

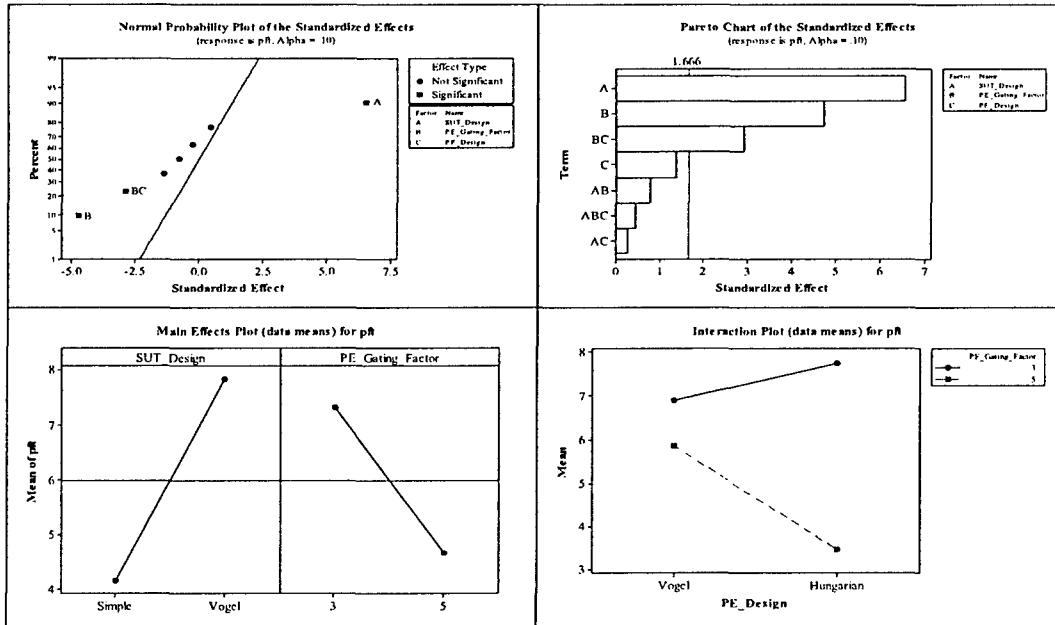
This section provides the Tier 0 DOE charts conducted in Section 9.1. The three factors SUT Design Gating Factor, PE Gating Factor and PE Design at two levels each are tested to find which of these factors affect the MOPs significantly. In Tier 0 we have three sensors on 2 platforms and they do not fuse any data within or across platform. Hence we have to only analyze track-to-truth associations for each of the MOPs. The summary of the results is shown in Table 4. Here for each MOP we have the Normal Probability plot and Pareto chart which summarizes the significant factors. Then for the significant factors we plot the main effects plot which tells us how the change in factor affects the MOP. For the significant interactions we plot the interaction plot which shows the effect of change in factor level combination on MOP.

After taking a look at the summary Table 4, we can say that SUT Design Gating Factor is comparatively more significant than PE Gating Factor and PE Design. SUT Design Gating Factor appears to be a significant factor in nearly all the Tier 0 DOE runs. So at Tier 0 we must be sensitive towards selection of SUT Design Gating Factor.

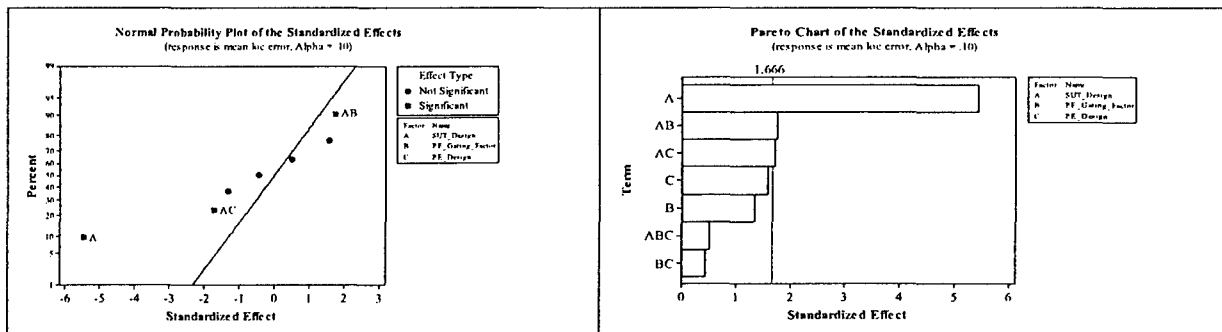
Radar track 1 to truth: Consistency

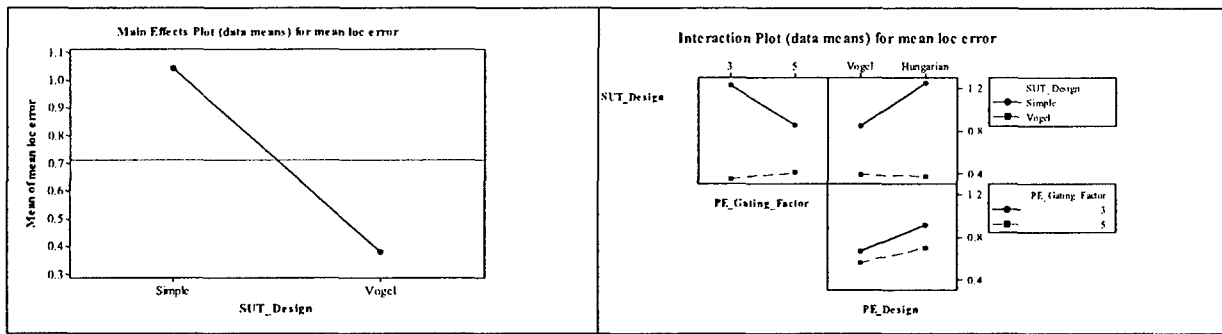


Radar track 1 to truth: Percentage of false tracks

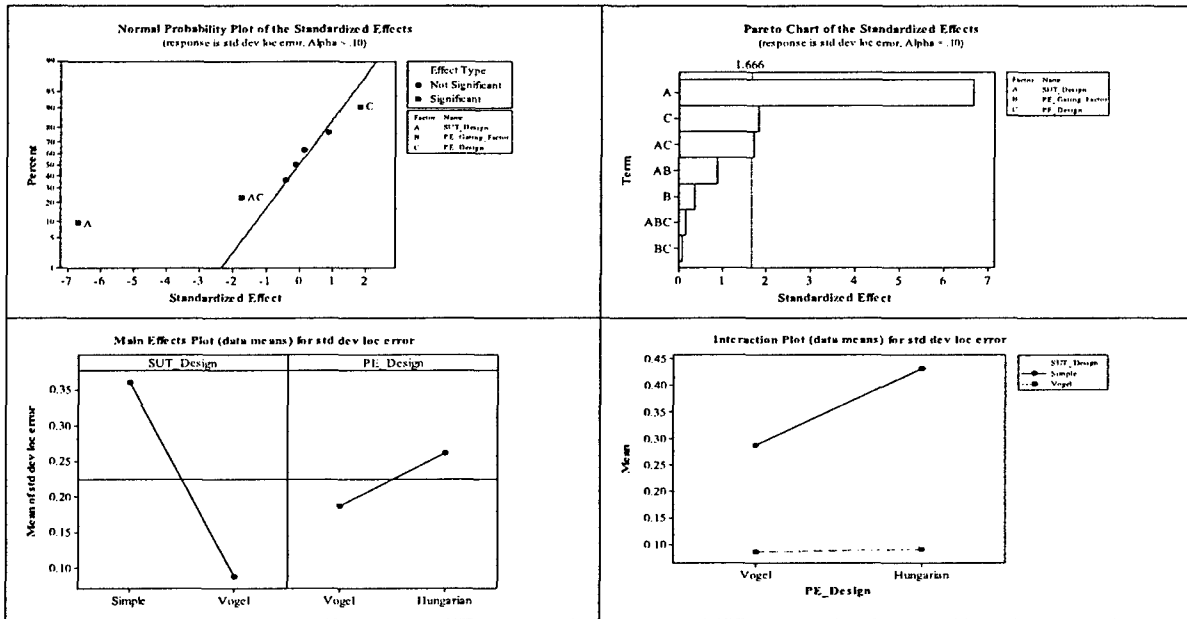


Radar track 1 to truth: Mean location error

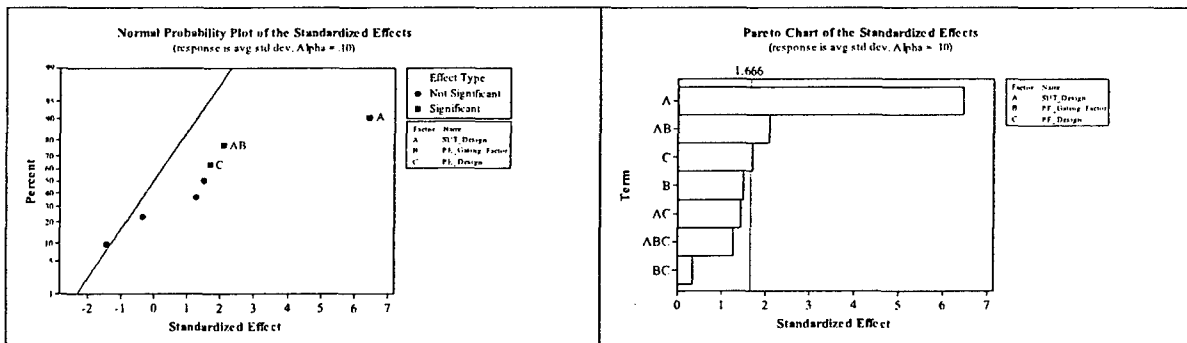


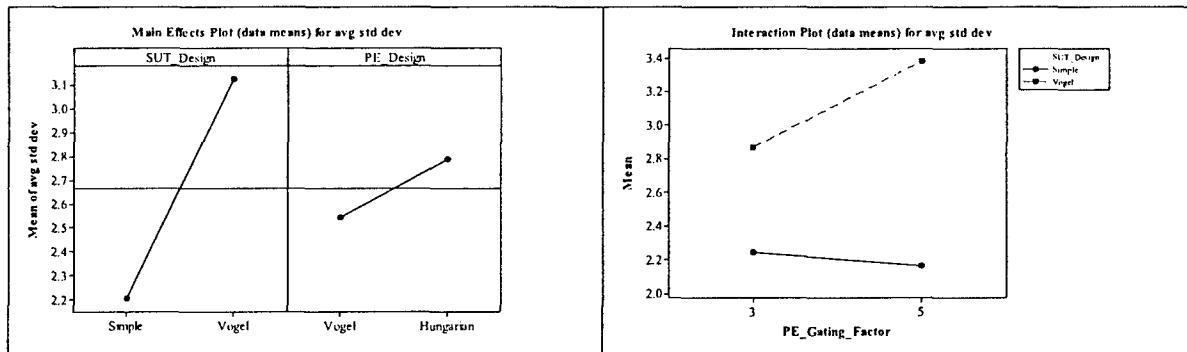


Radar track 1 to truth: Average standard deviation for location error

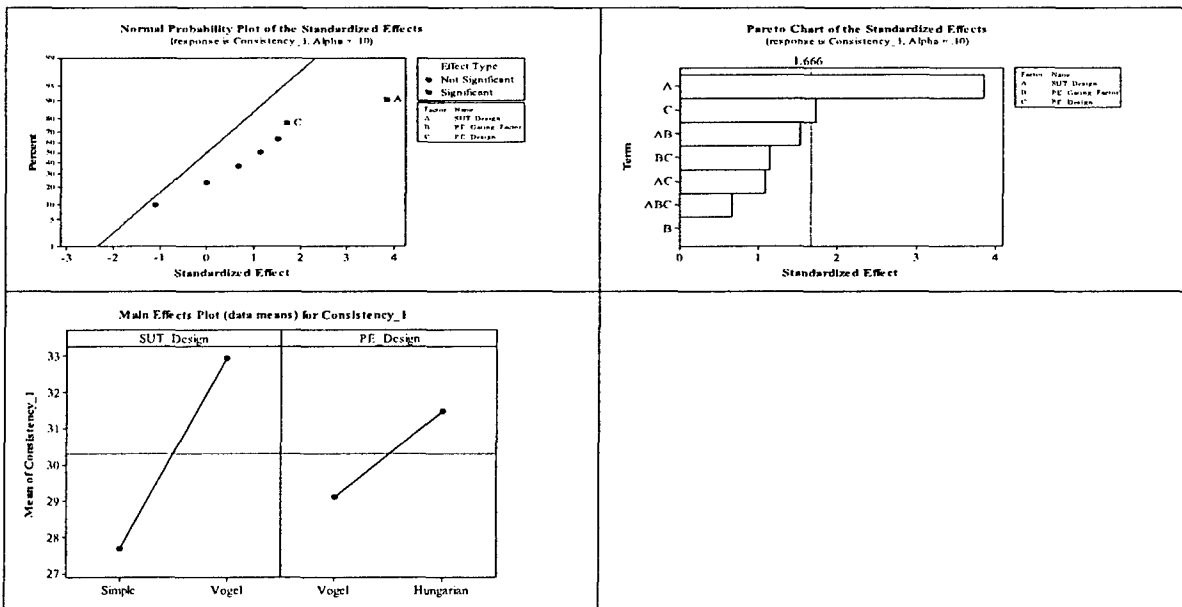


Radar track 1 to truth: Average standard deviation

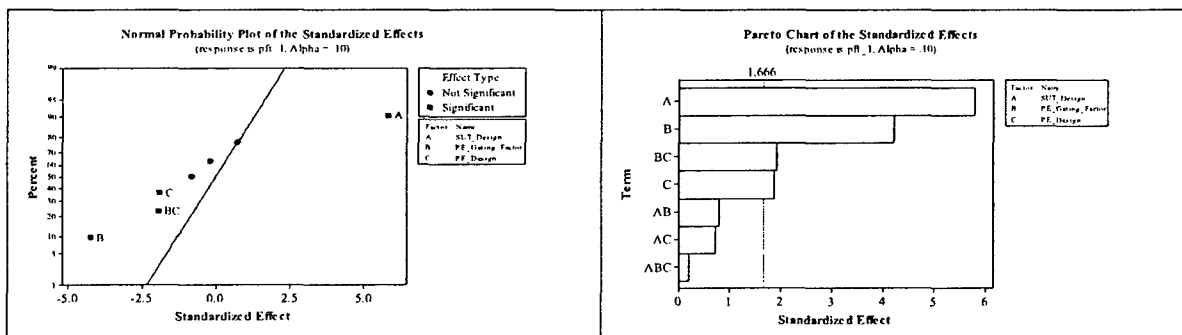


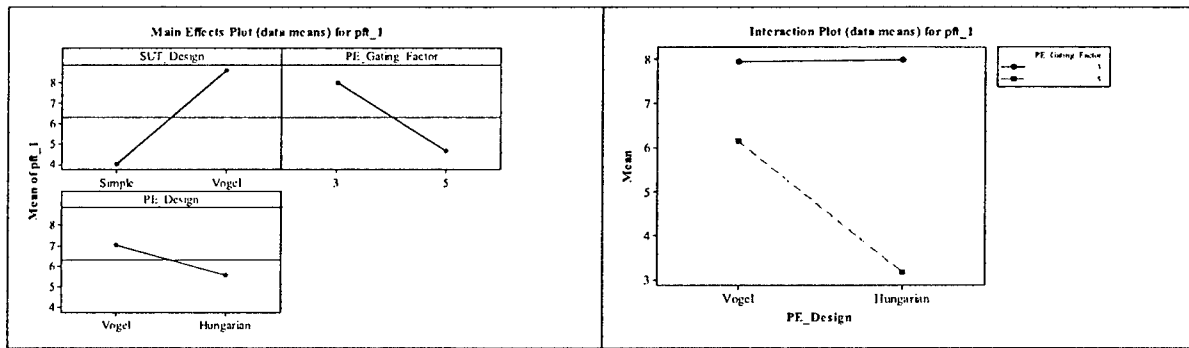


Radar track 2 to truth: Consistency

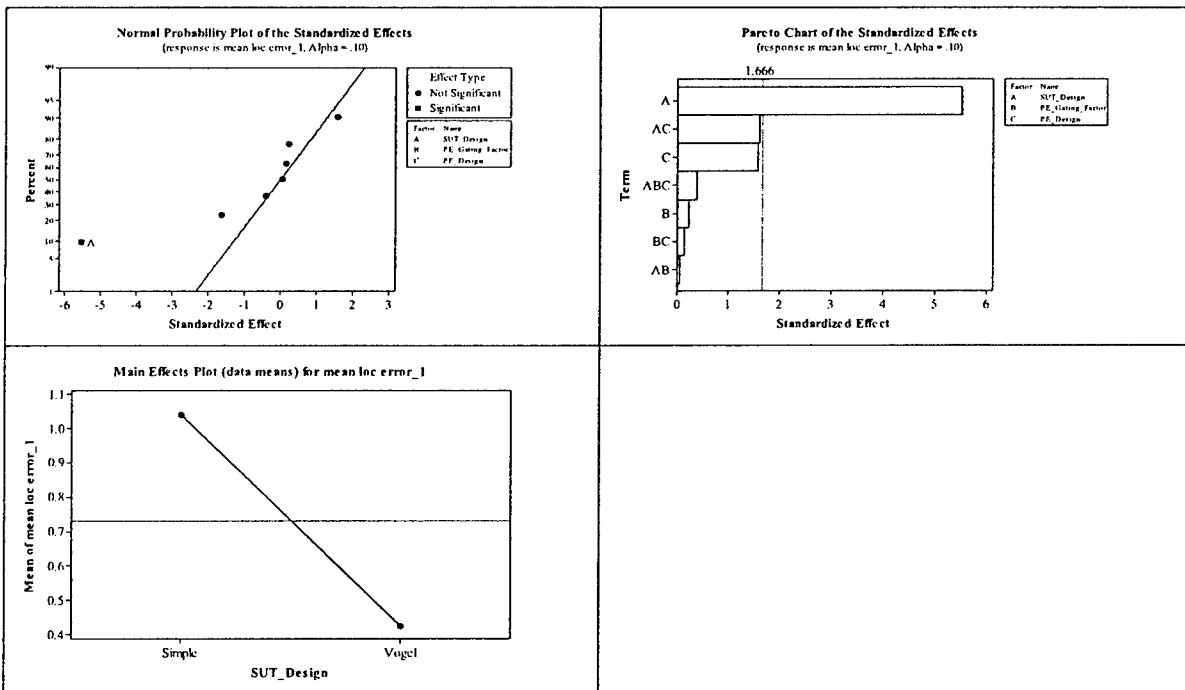


Radar track 2 to truth: Percentage of false tracks

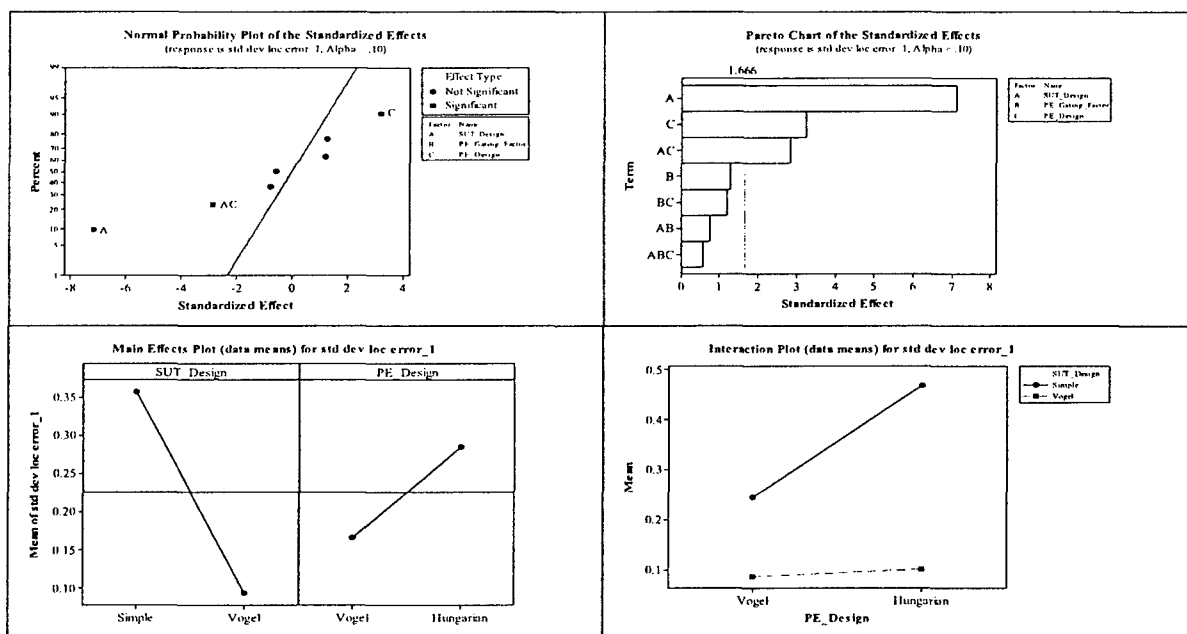




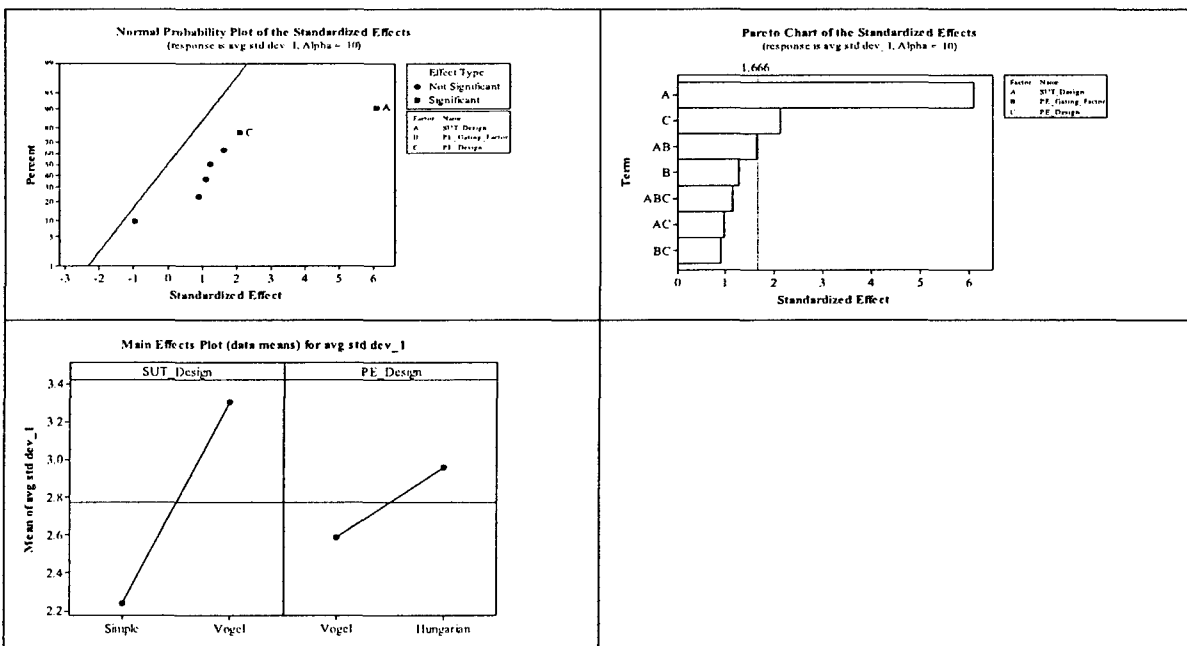
Radar track 2 to truth: Mean location error



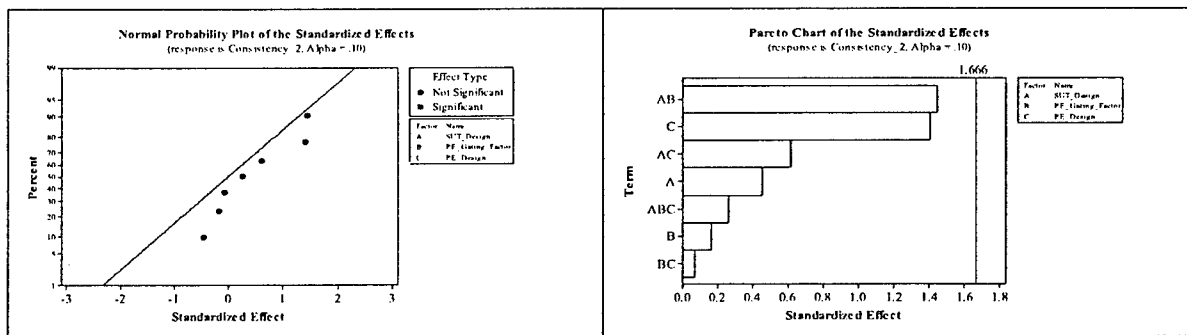
Radar track 2 to truth: Average standard deviation of location error



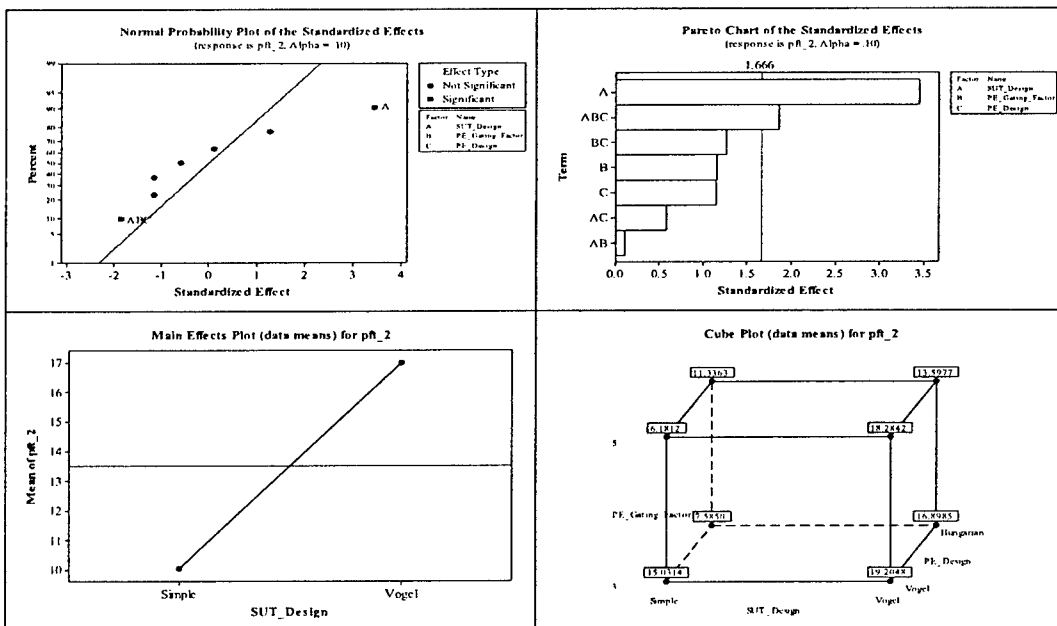
Radar track 2 to truth: Average standard deviation



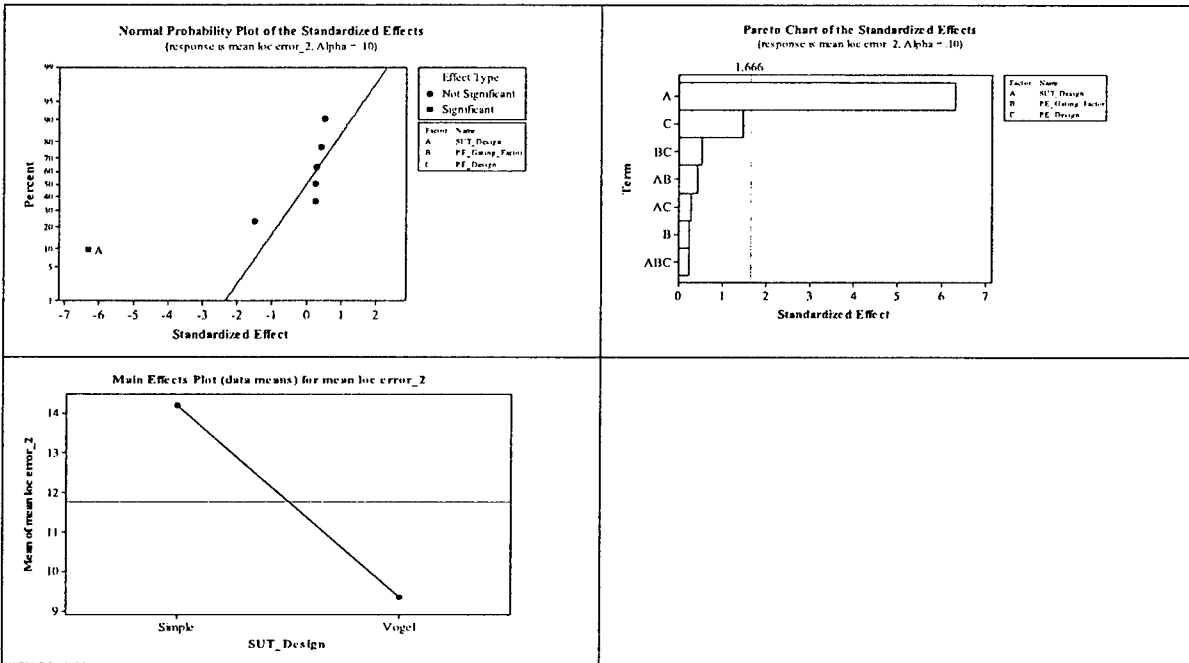
ESM track 1 to truth: Consistency



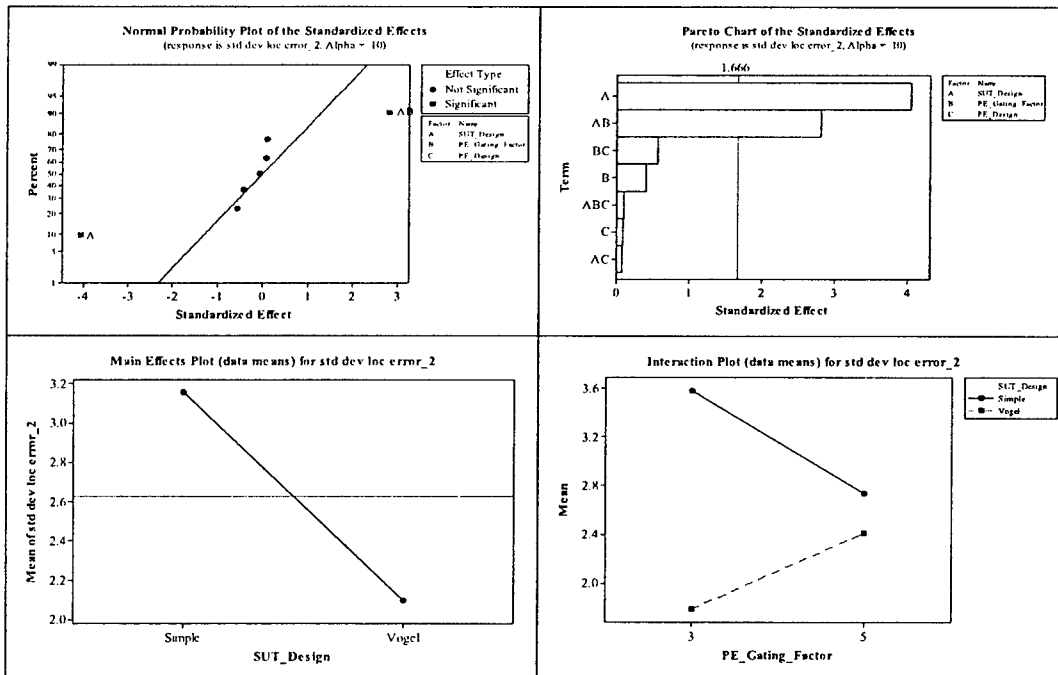
ESM track 1 to truth: Percentage of false tracks



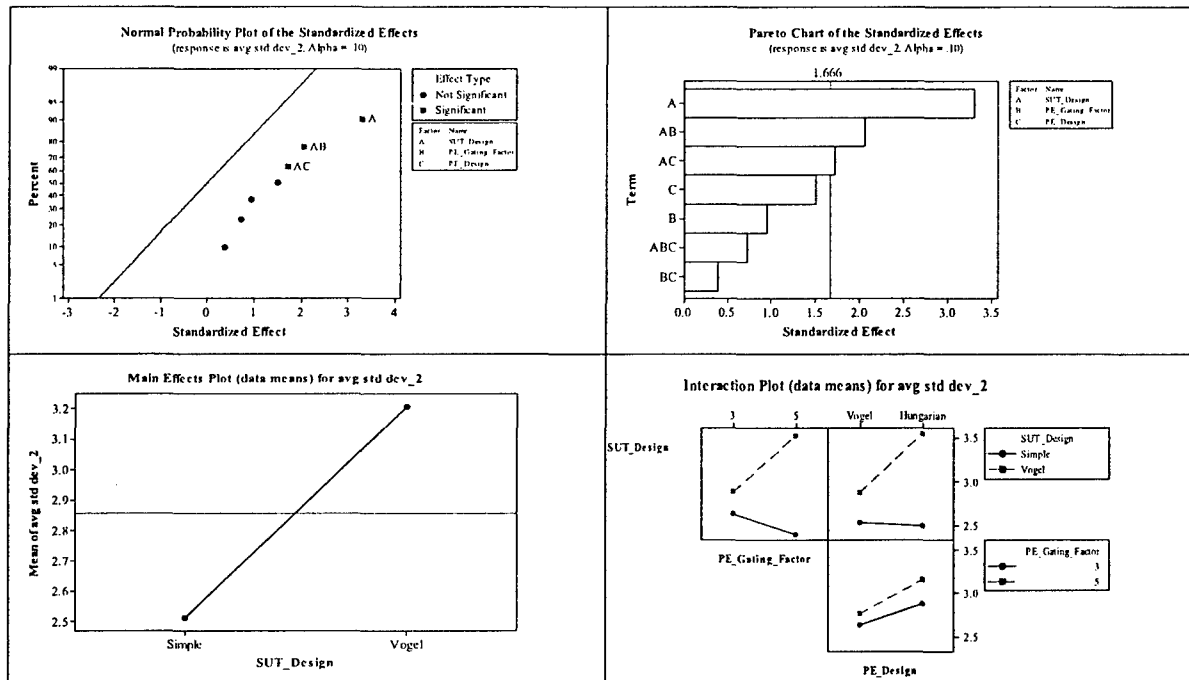
ESM track 1 to truth: Mean Location Error



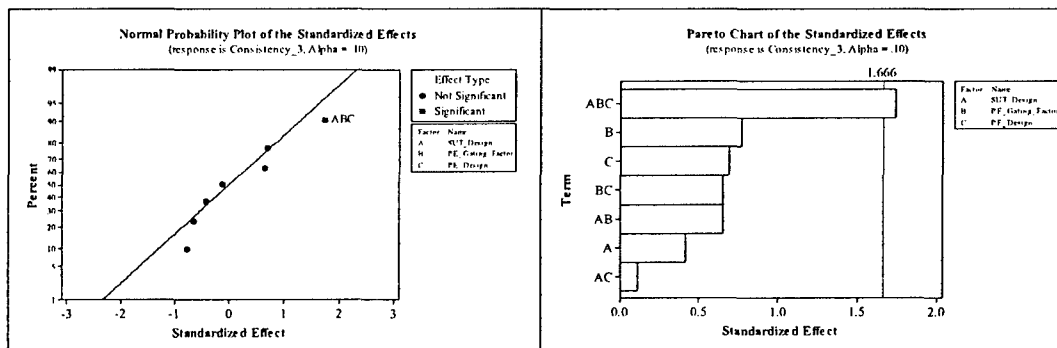
ESM track 1 to truth: Average standard deviation of location error

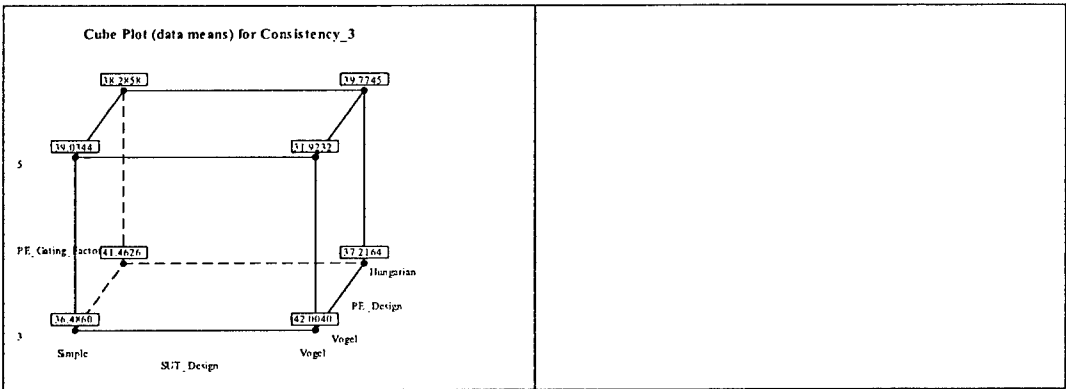


ESM track 1 to truth: Average standard deviation

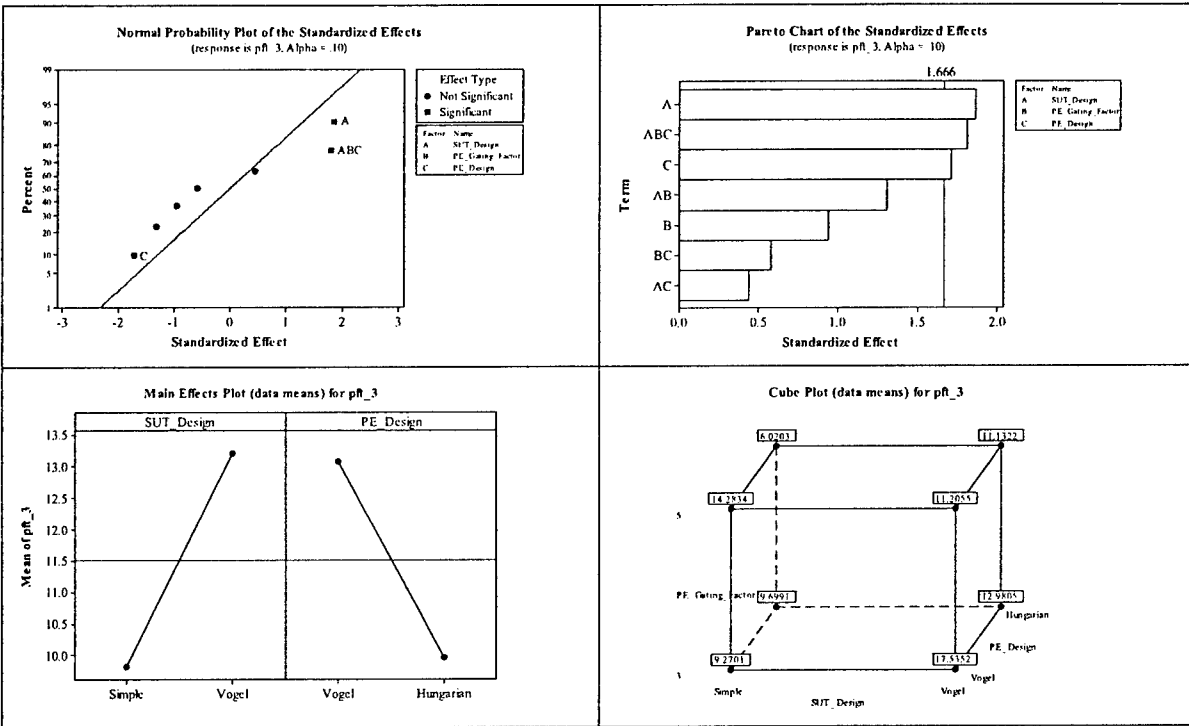


ESM track 2 to truth: Consistency

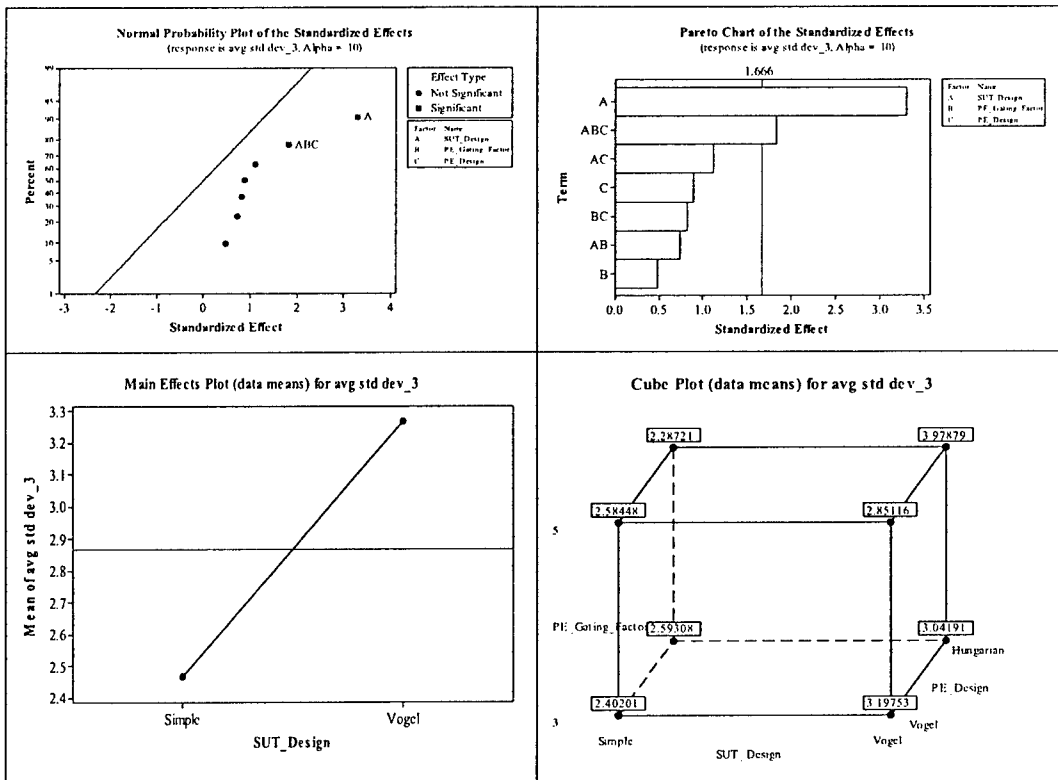




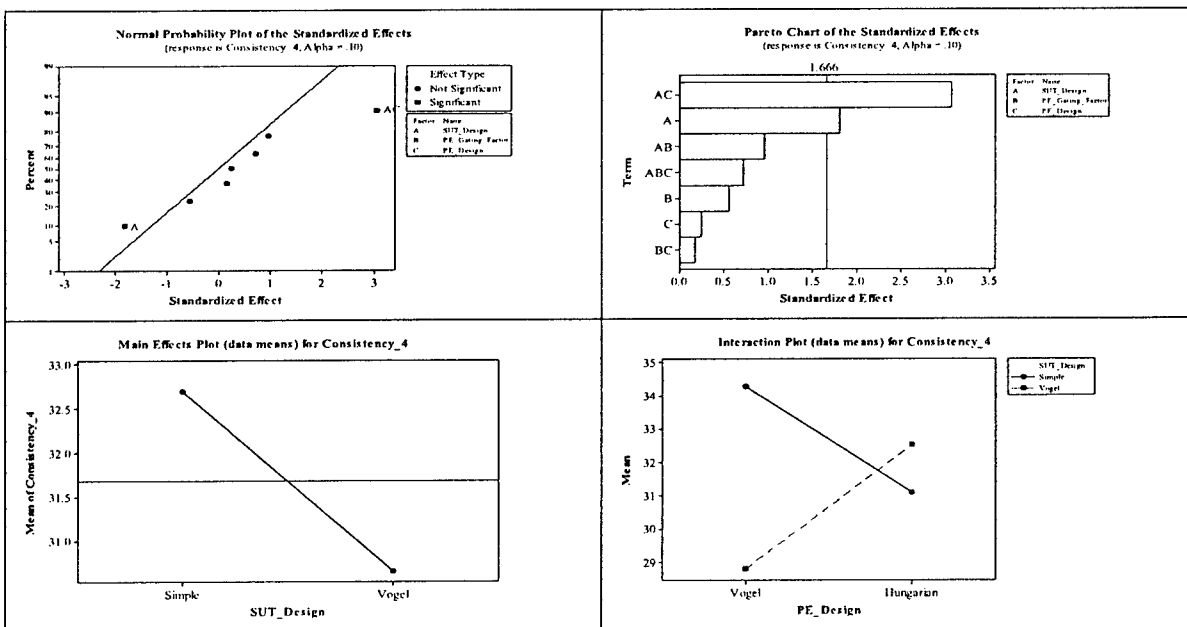
ESM track 2 to truth: Percentage of false tracks



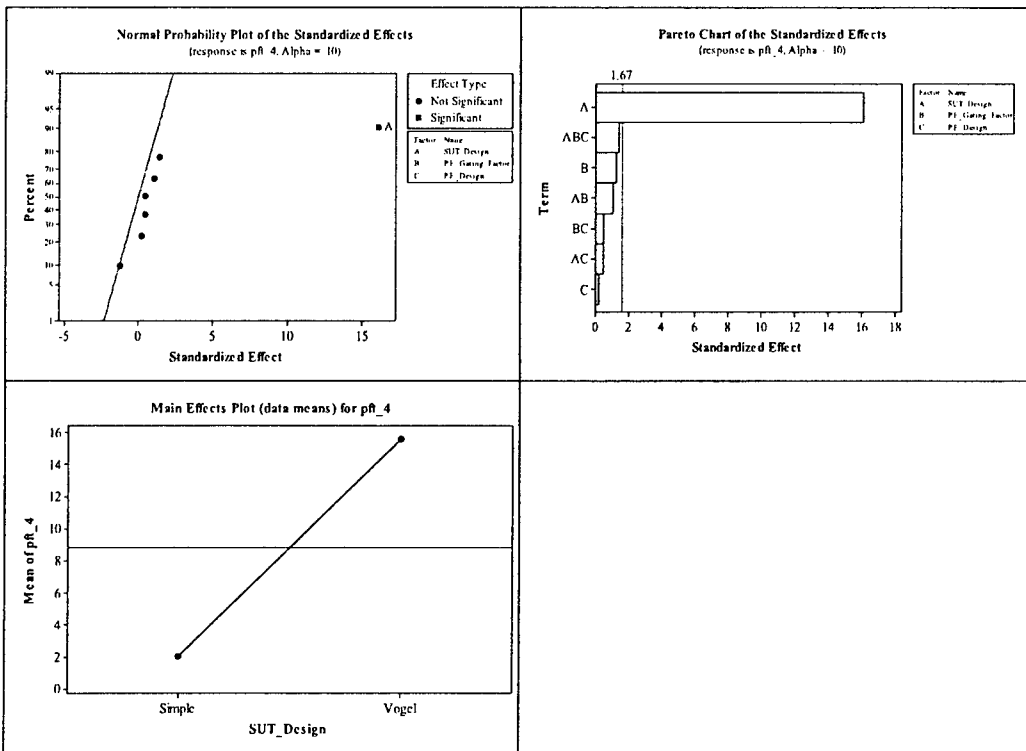
ESM track 2 to truth: Mean location error



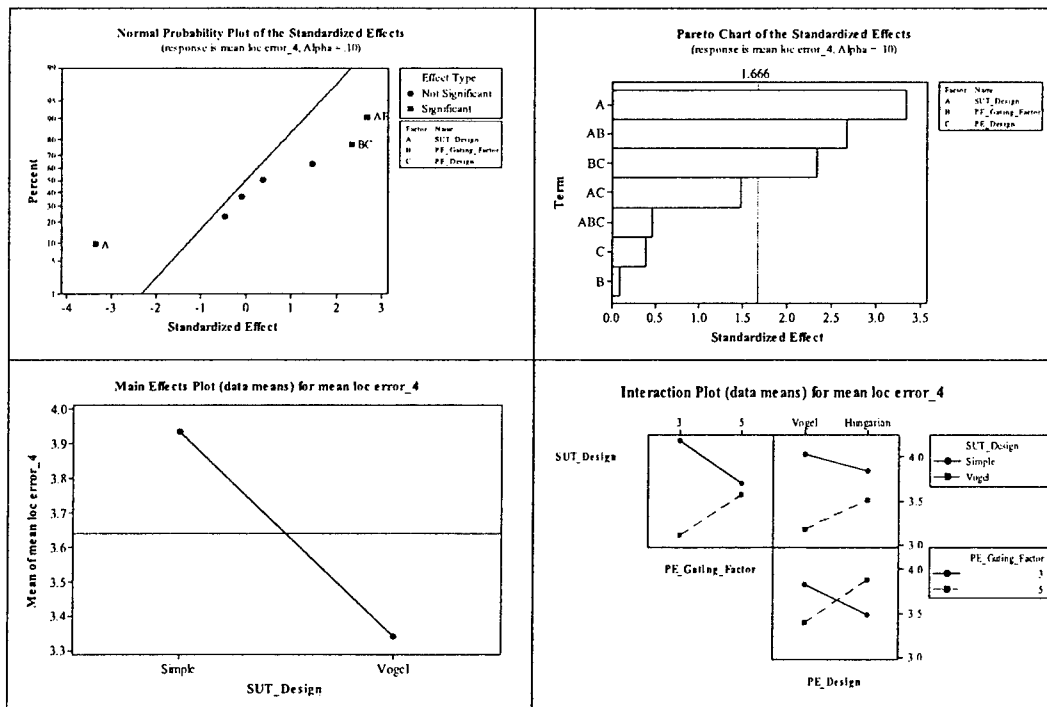
IRST track 1 to truth: Consistency



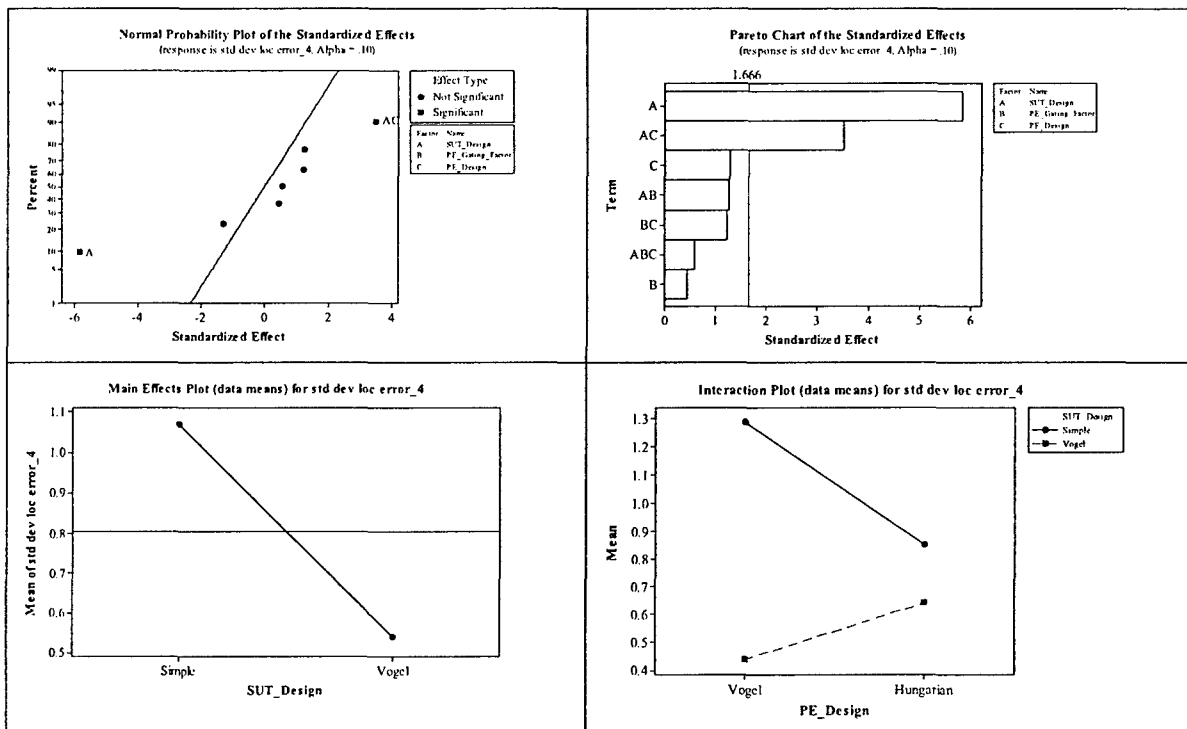
IRST track 1 to truth: Percentage of false tracks



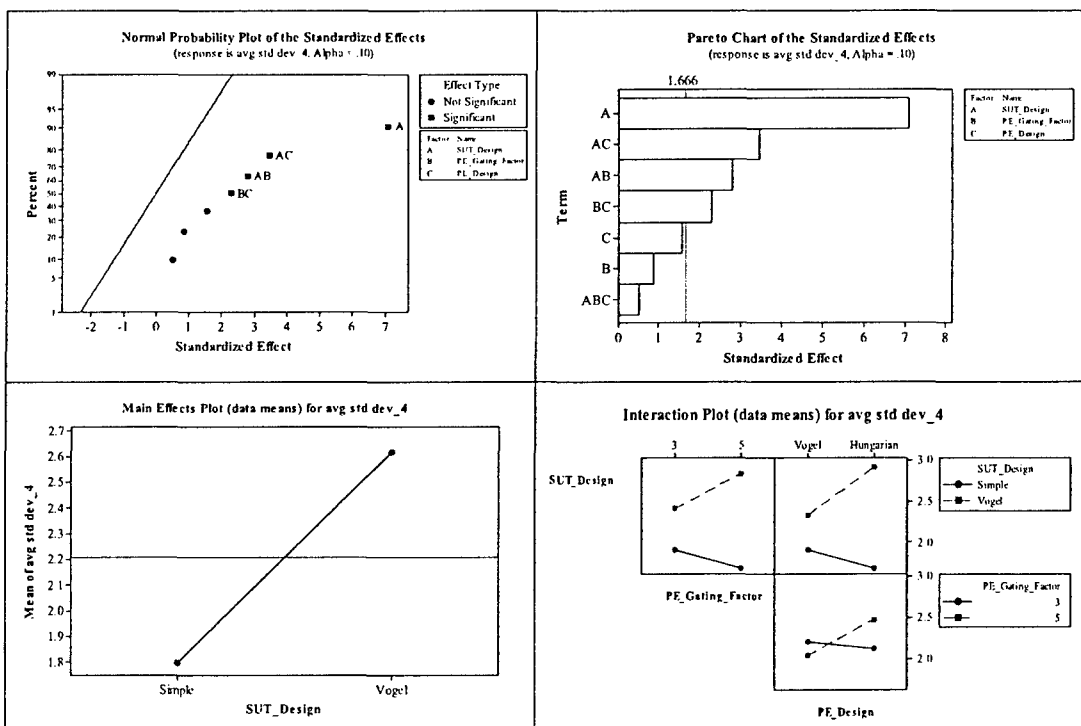
IRST track 1 to truth: Mean location error



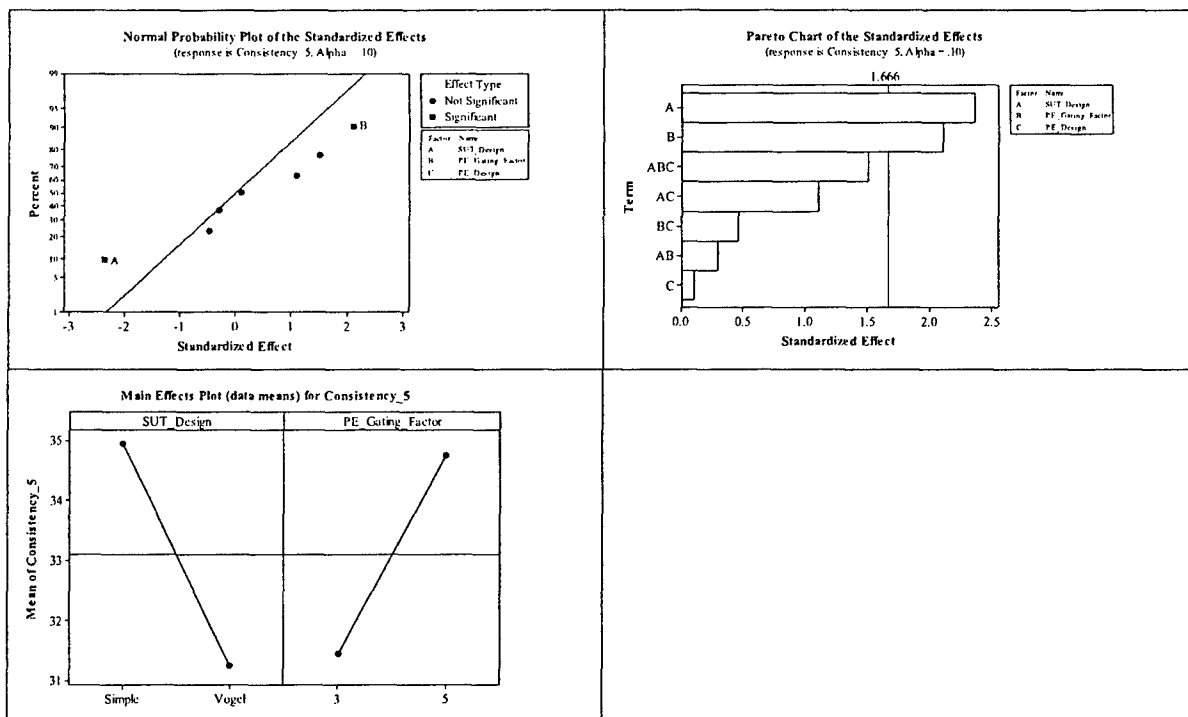
IRST track 1 to truth: Average standard deviation of location error



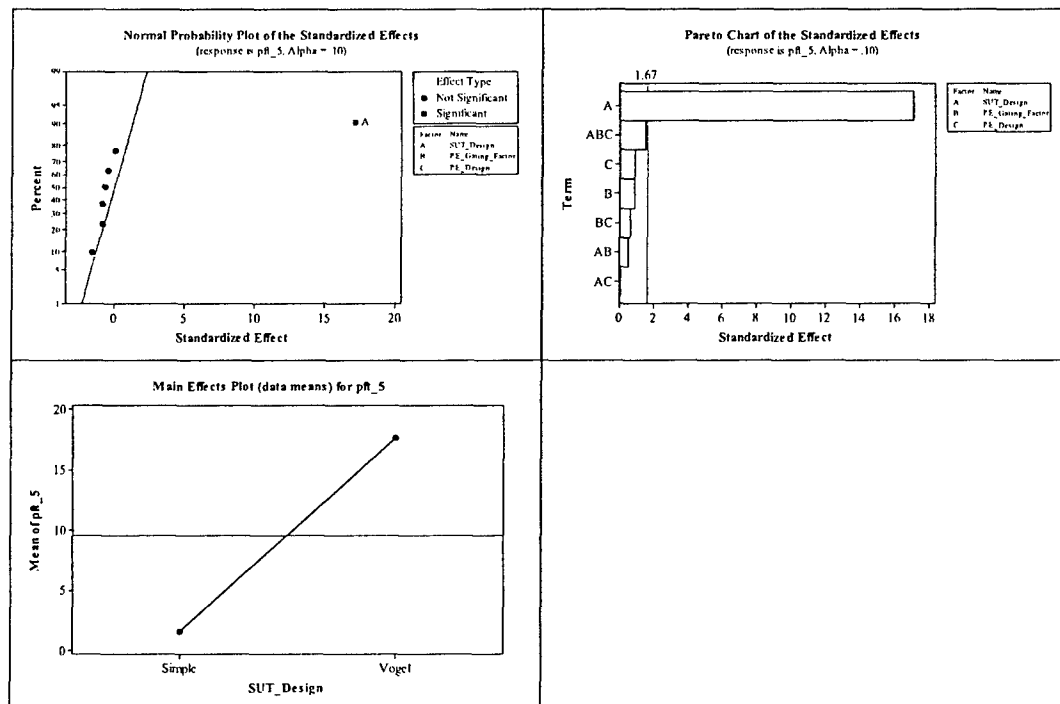
IRST track 1 to truth: Average standard deviation



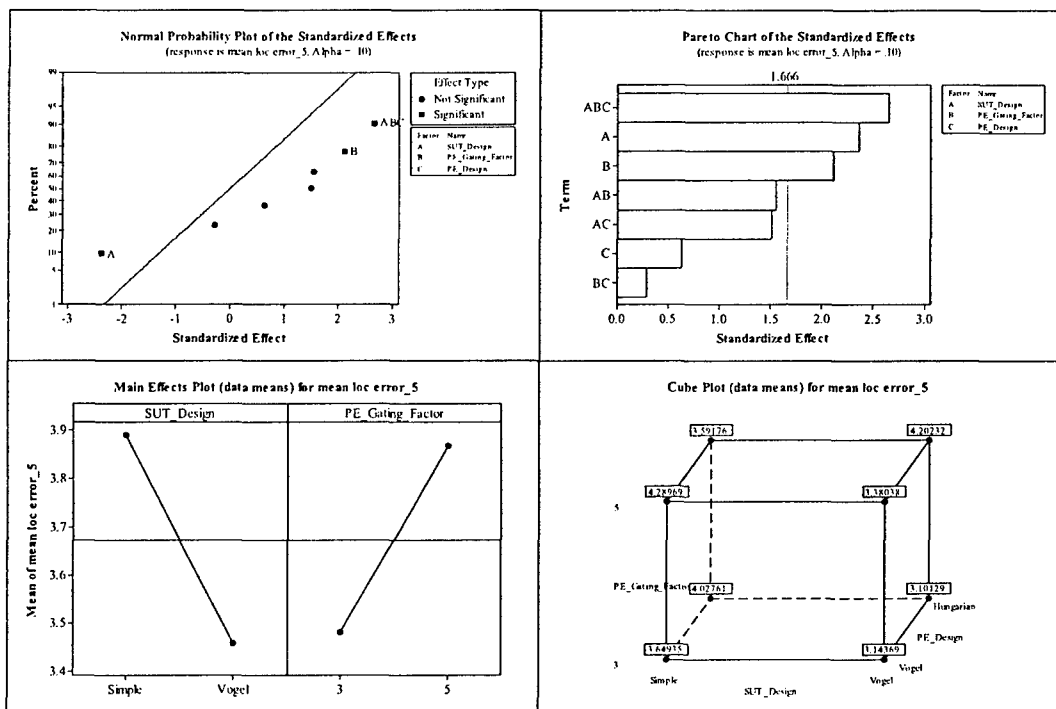
IRST track 2 to truth: Consistency



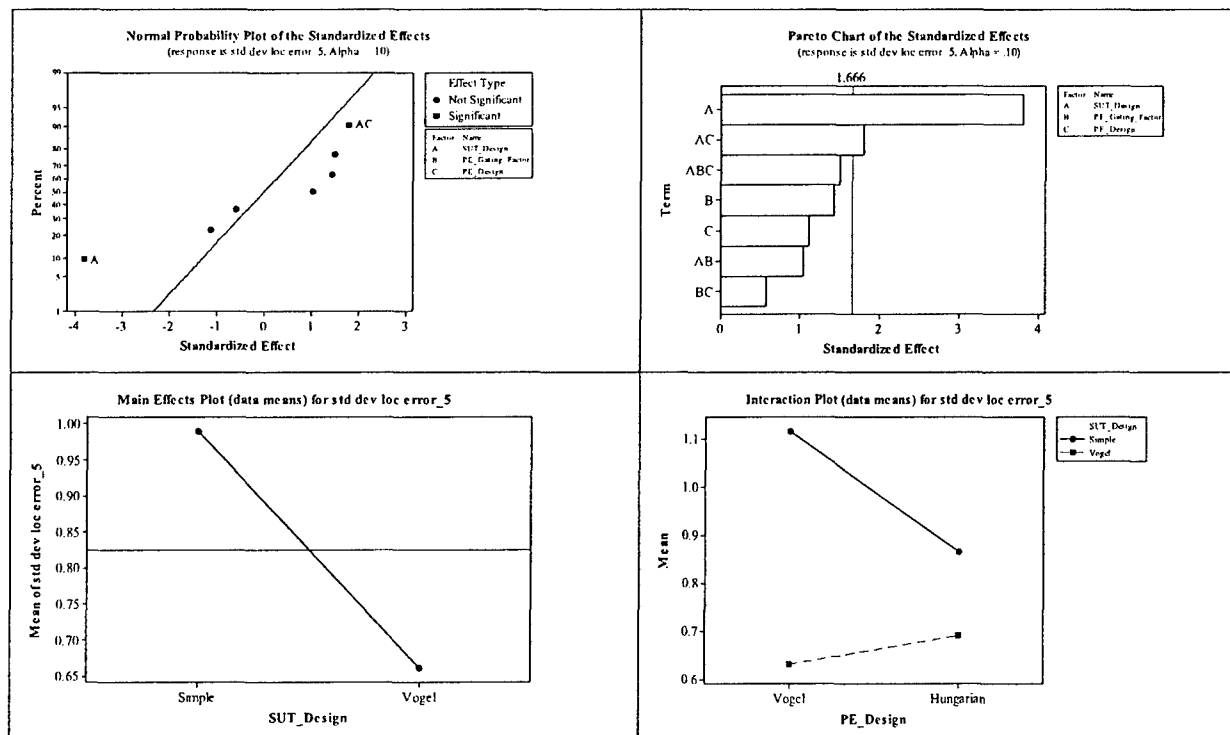
IRST track 2 to truth: Percentage of false tracks



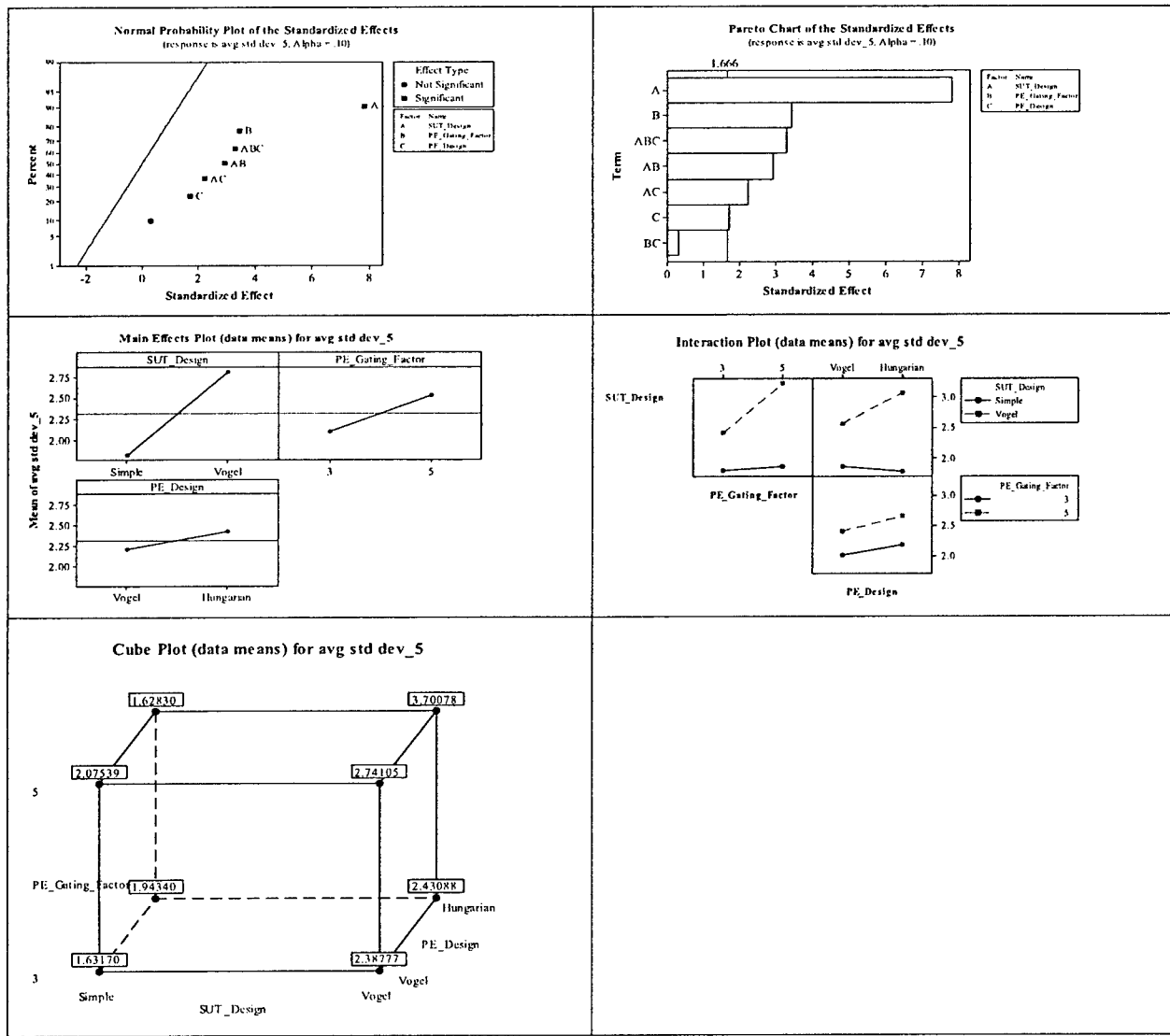
IRST track 2 to truth: Mean location error



IRST track 2 to truth: Average standard deviation of location error



IRST track 2 to truth: Average standard deviation

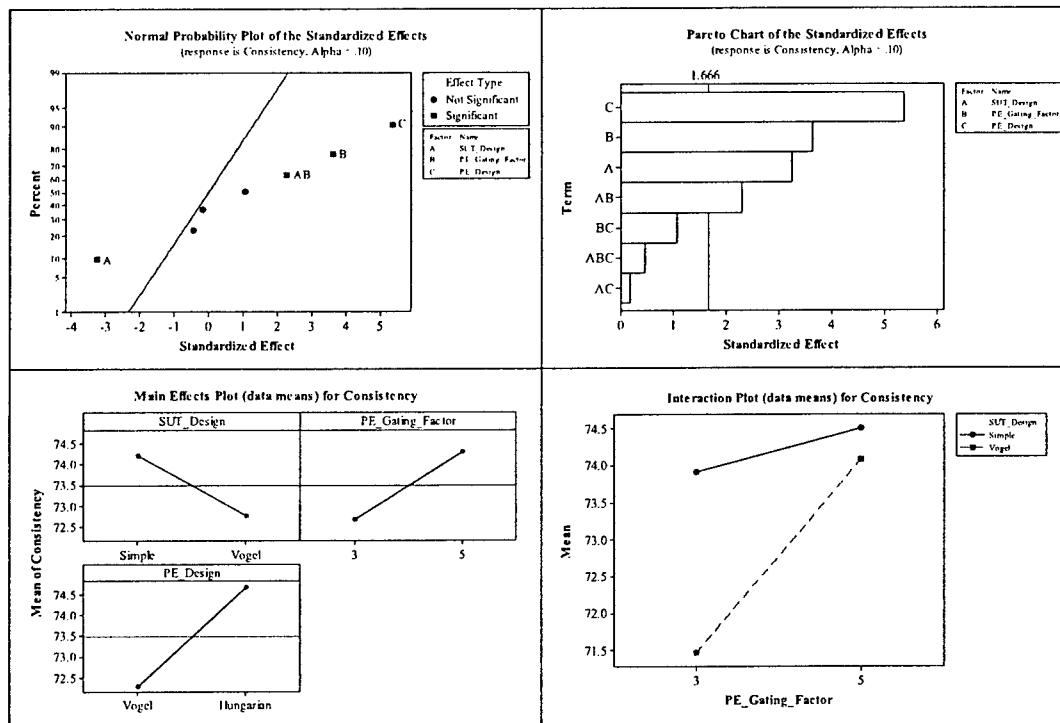


D. Tier 1 DOE Charts

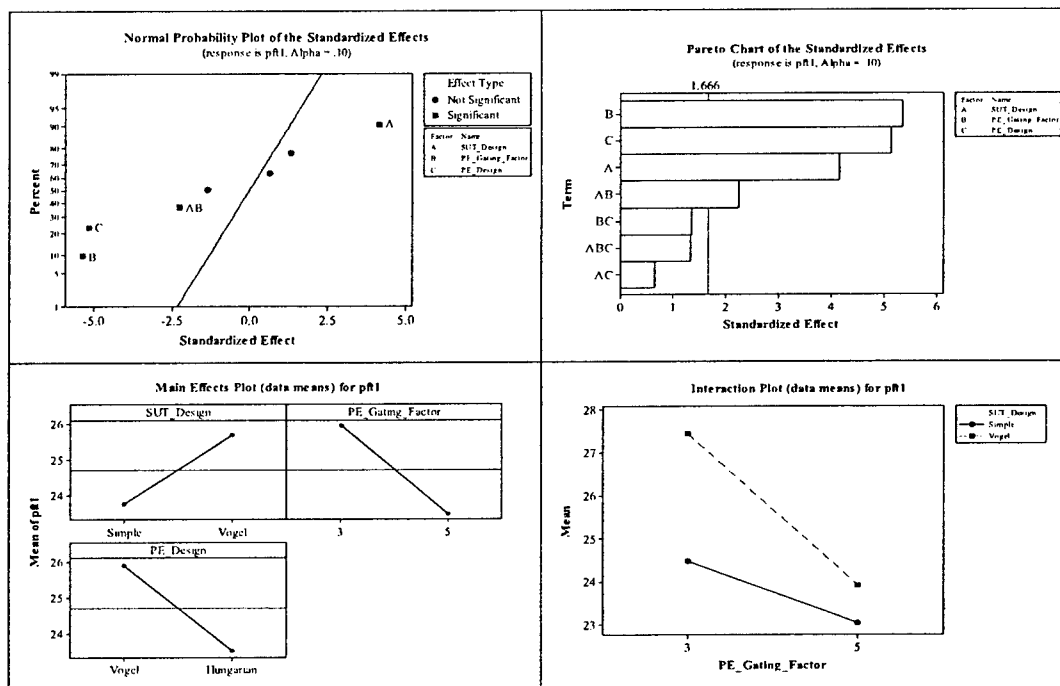
This section provides the Tier 1 DOE charts conducted in Section 9.1. The three factors SUT Design Gating Factor, PE Gating Factor and PE Design at two levels each are tested to find which of these factors affect the MOPs significantly. In Tier 1 we have three sensors on 2 platforms and they fuse data within platform (not across platform). So we have to analyze track-to-truth and track-to-track associations for each of the MOPs. The summary of the results is shown in Table 5. Here for each MOP we have the Normal Probability plot and Pareto chart which summarizes the significant factors. Then for the significant factors we plot the main effects plot which tells us how the change in factor affects the MOP. For the significant interactions we plot the interaction plot which shows the effect of change in factor level combination on MOP.

After taking a look at the summary Table 5, we can say that all the three factors SUT Design Gating Factor, PE Gating Factor and PE Design are very significant. All the three factors appear to be significant in nearly all the Tier 1 DOE runs. The interaction between SUT Design Gating Factor and PE Gating Factor is mostly significant for all the MOPs.

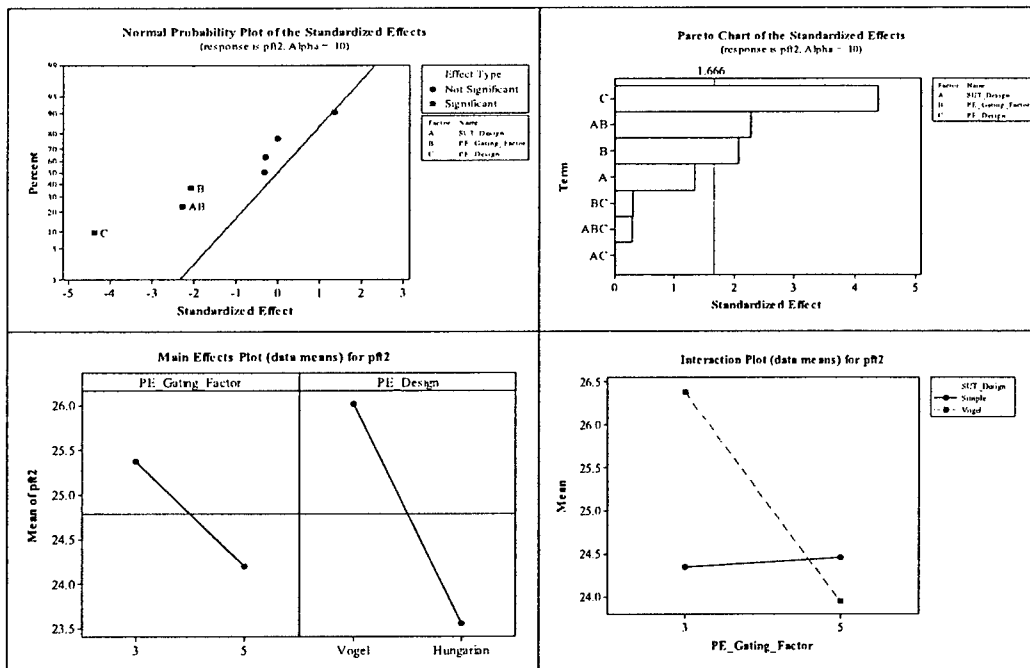
Track to track: Consistency



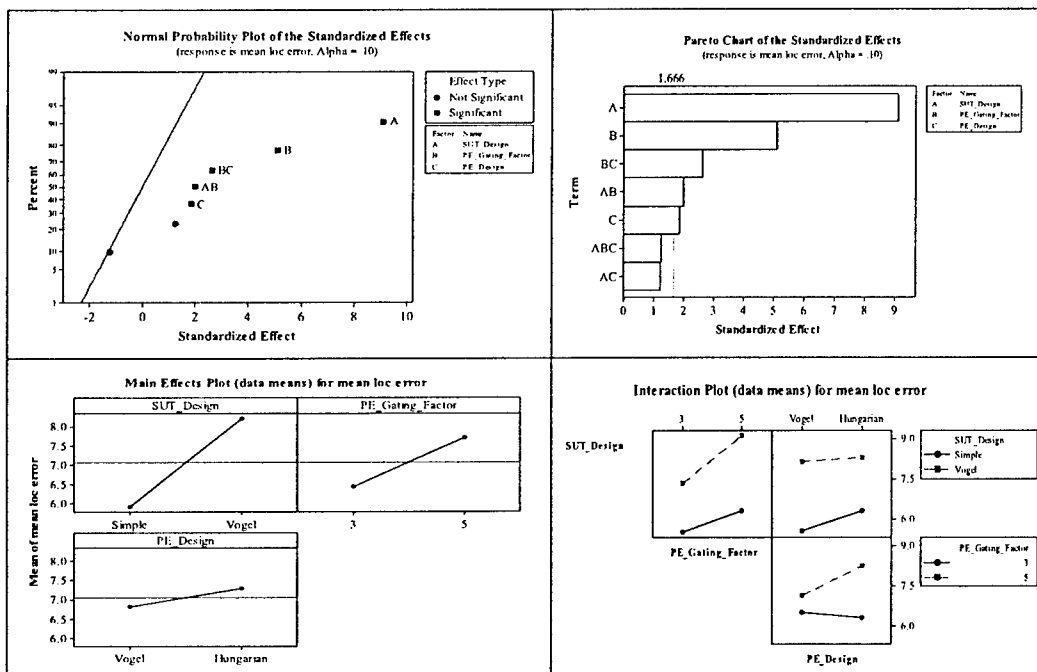
Track to track: Percentage of false tracks in track 1



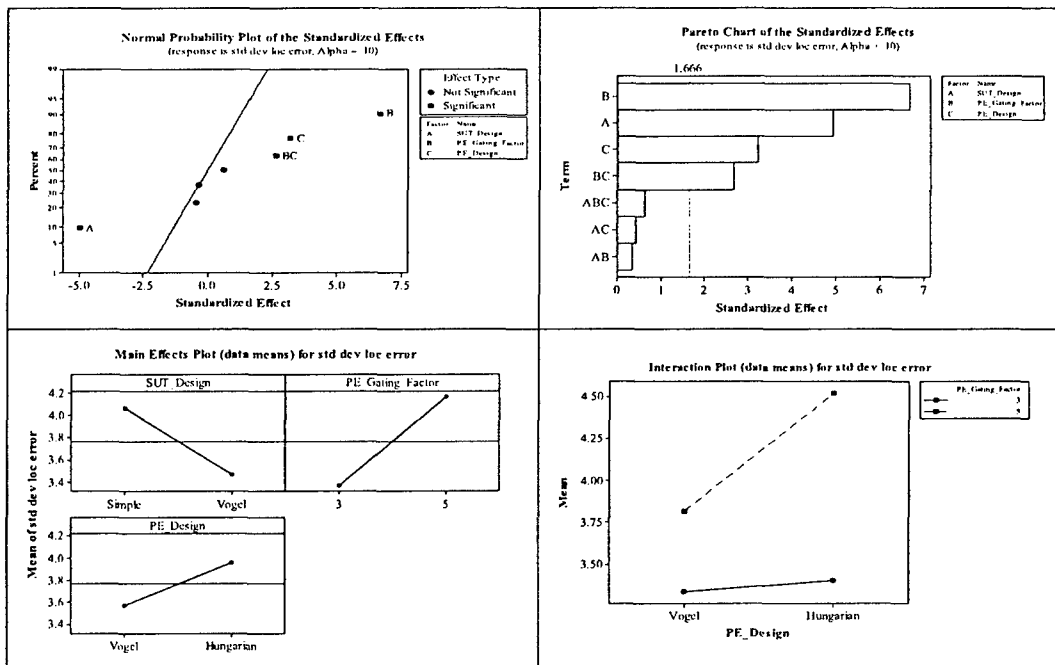
Track to track: Percentage of false tracks in track 2



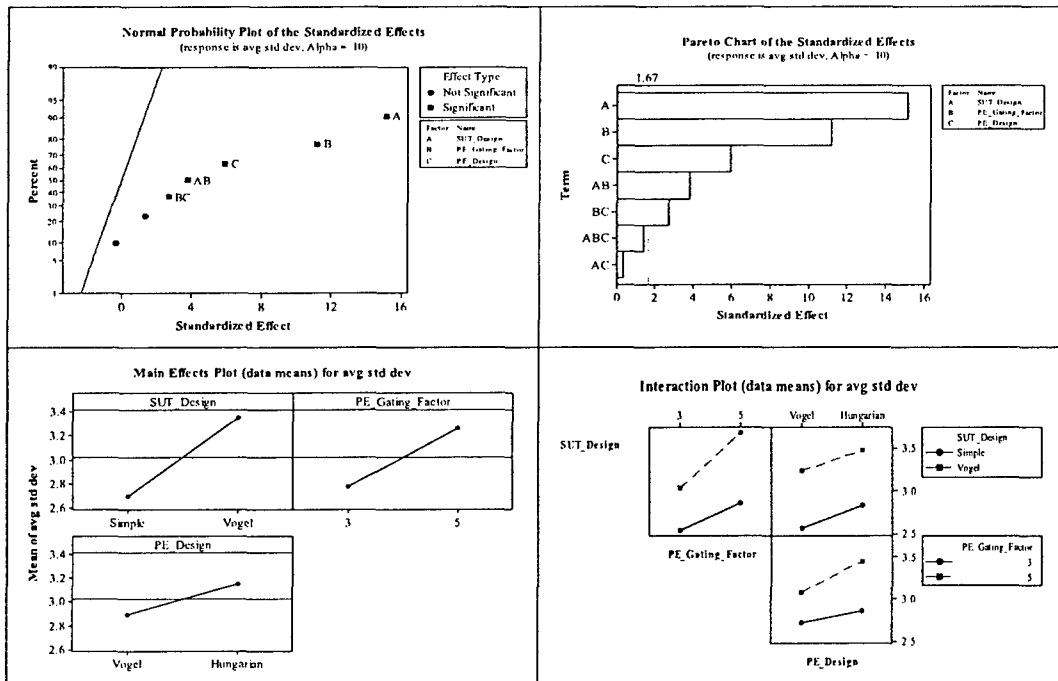
Track to track: Mean location error



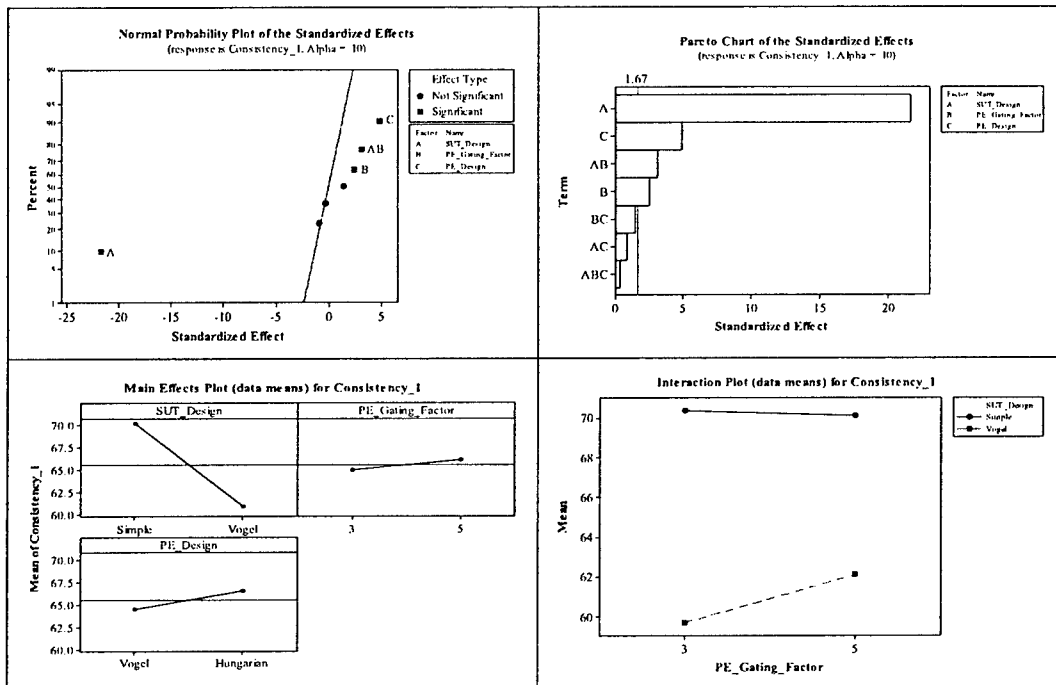
Track to track: Average standard deviation of location error



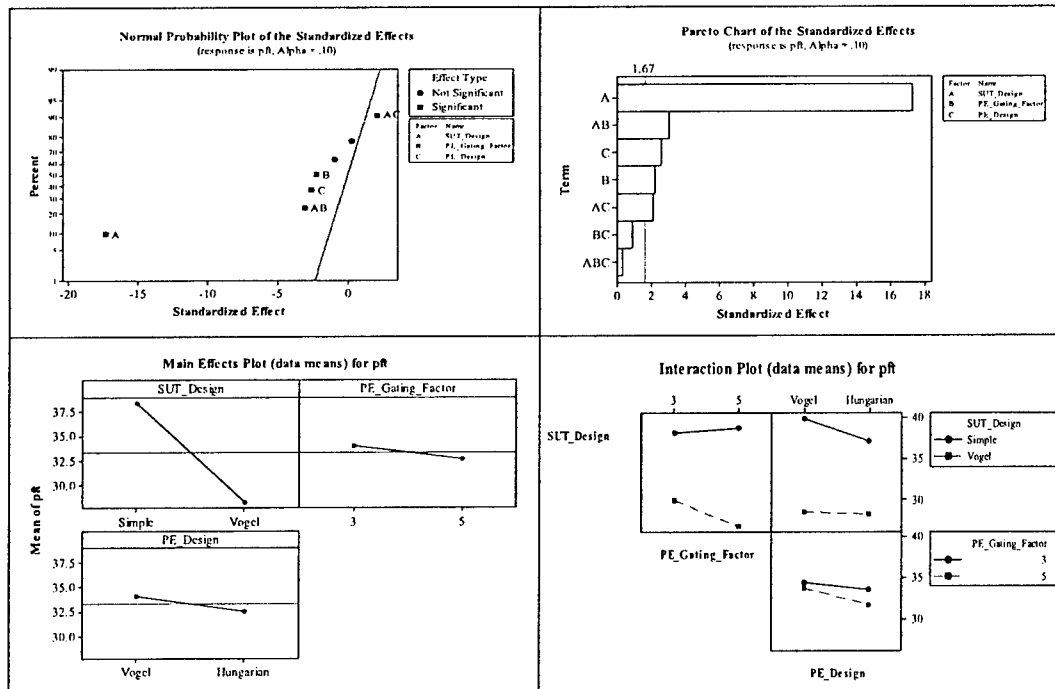
Track to track: Average standard deviation



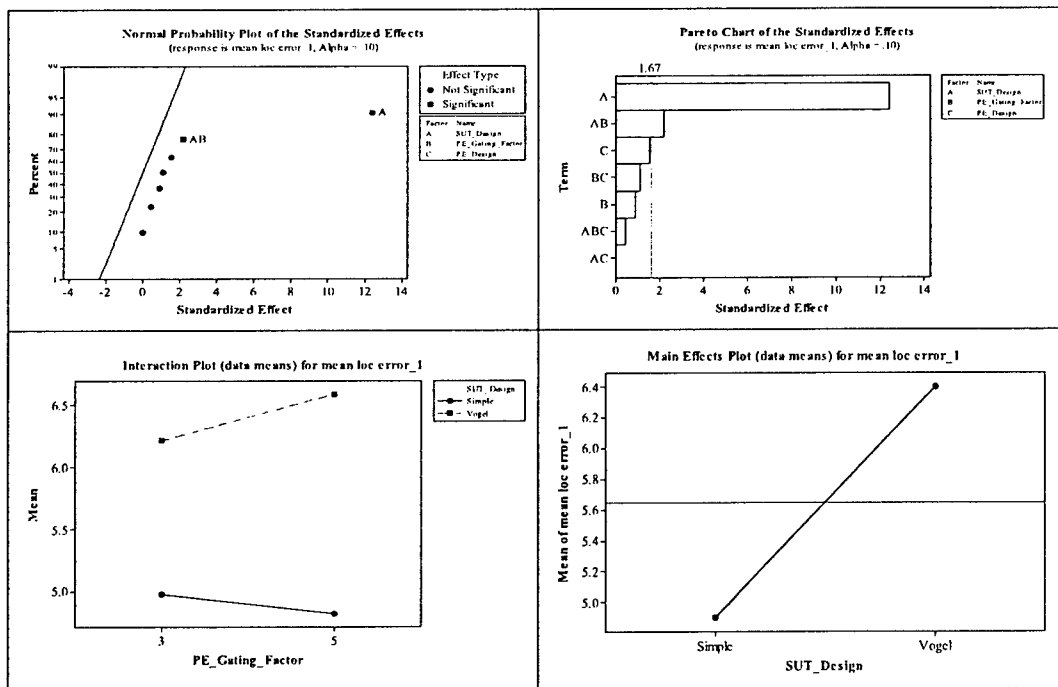
Track 1 to truth: Consistency



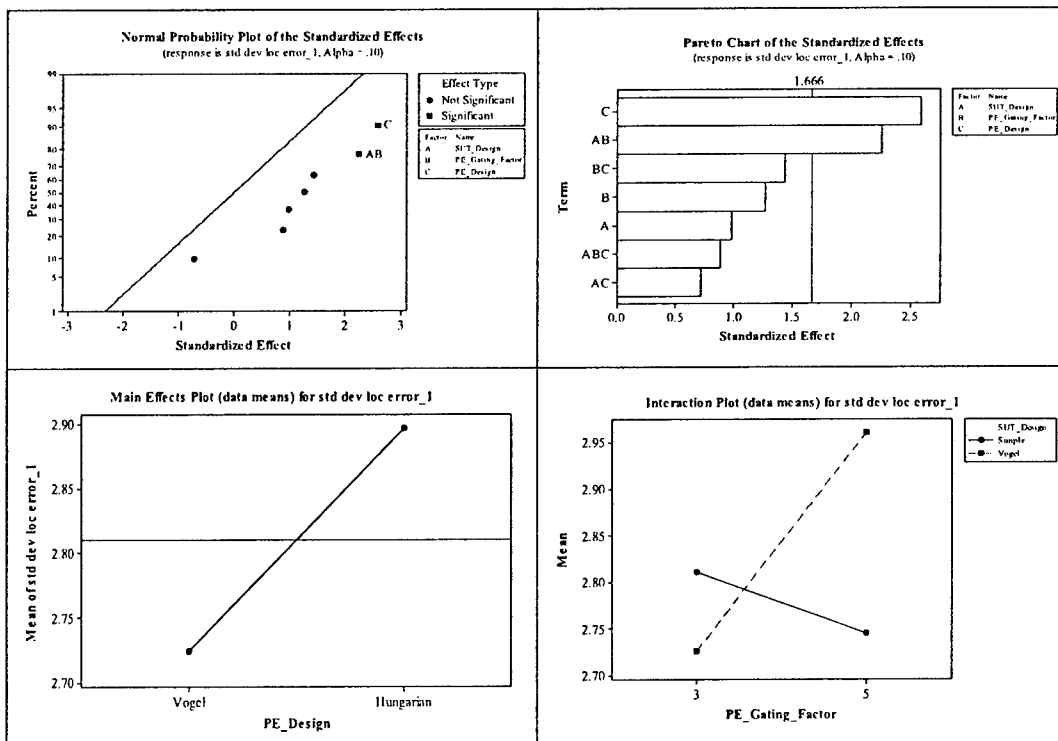
Track 1 to truth: Percentage of false tracks



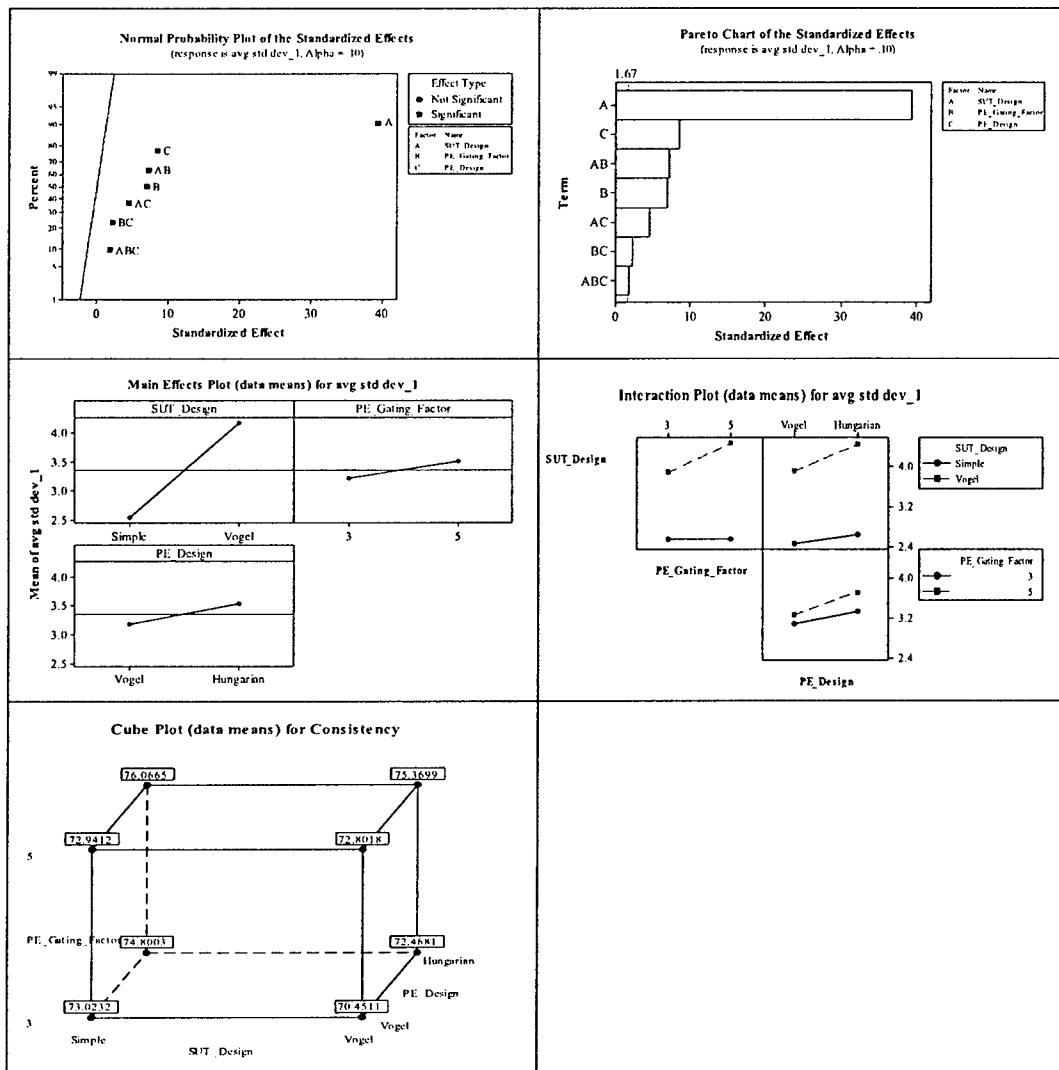
Track 1 to truth: Mean location error



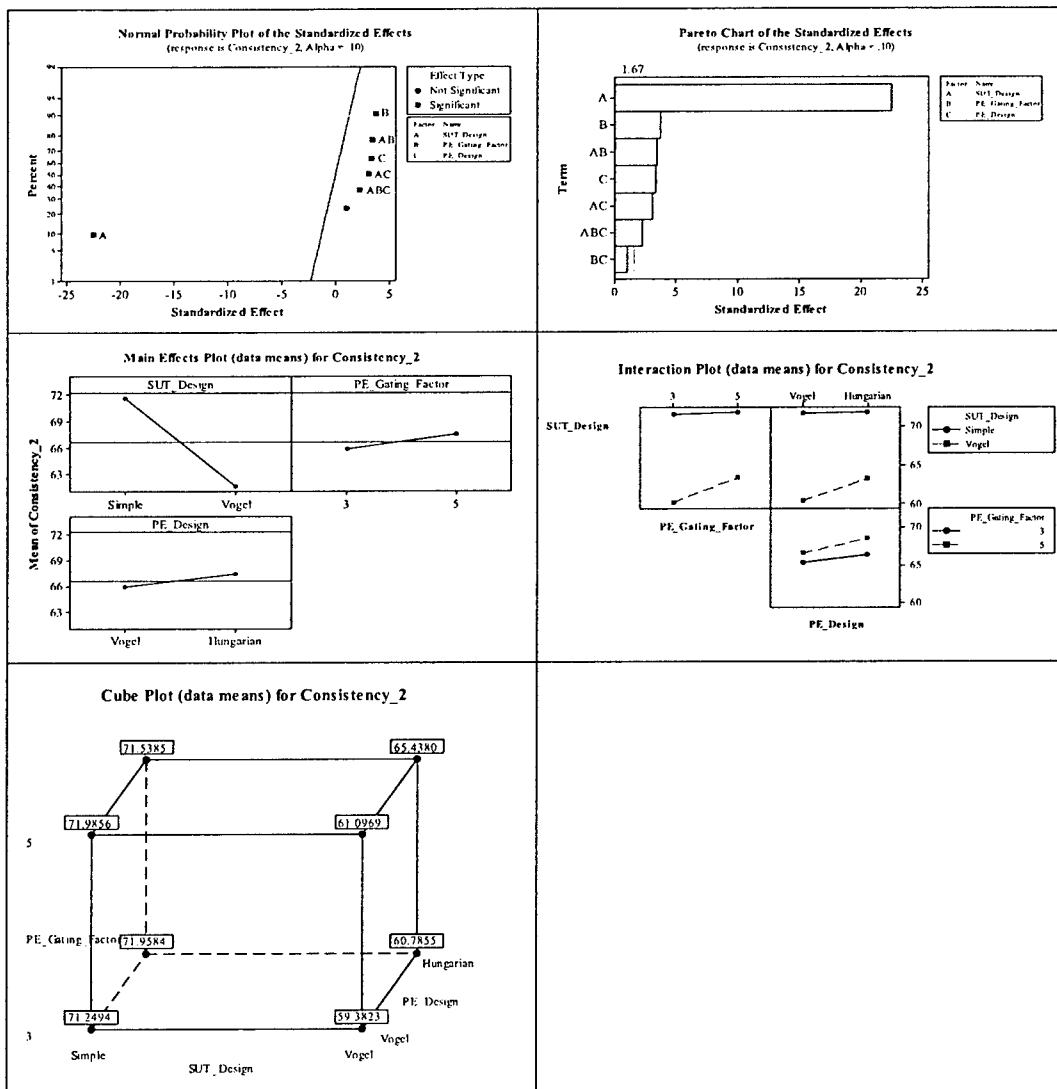
Track 1 to truth: Average standard deviation of location error



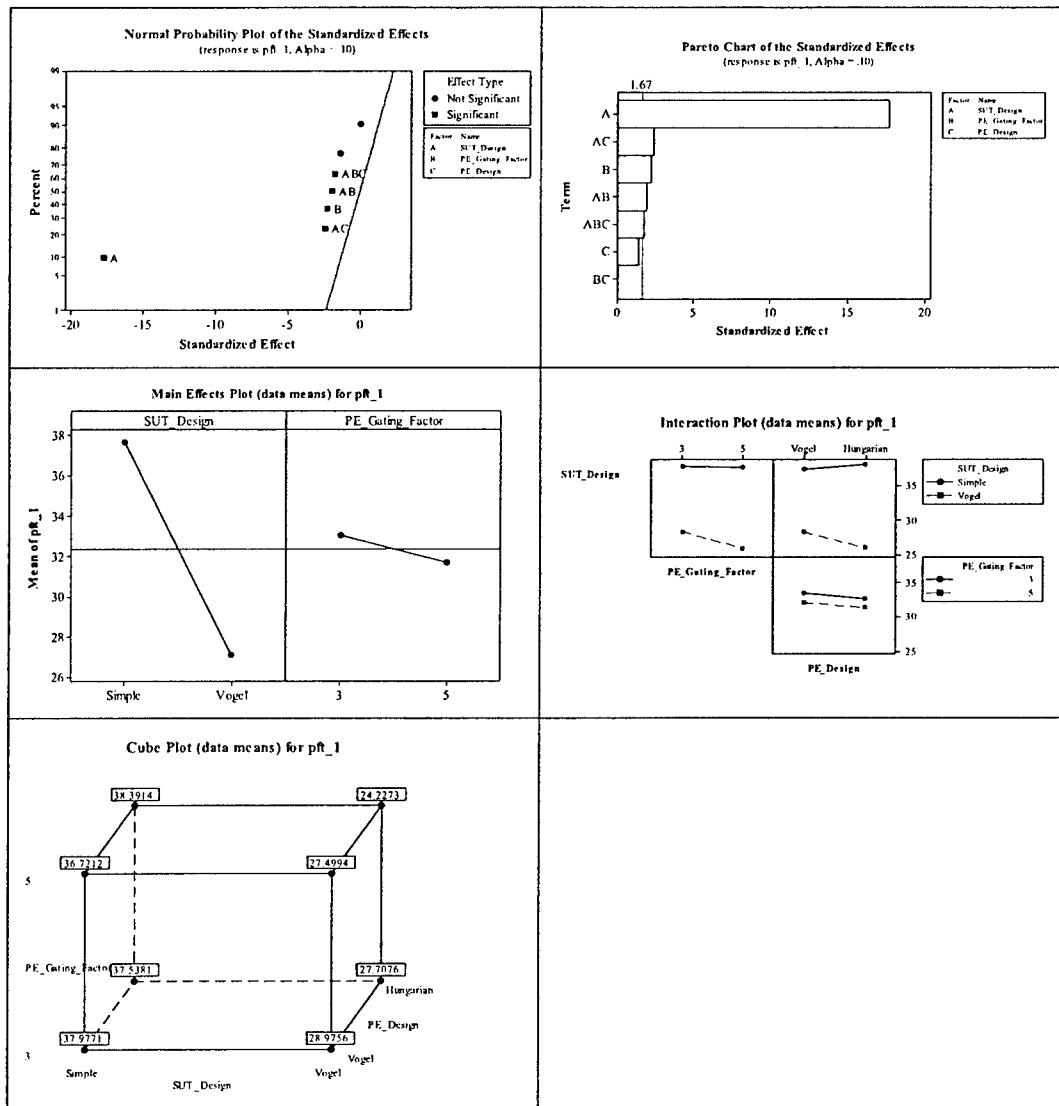
Track 1 to truth: Average standard deviation



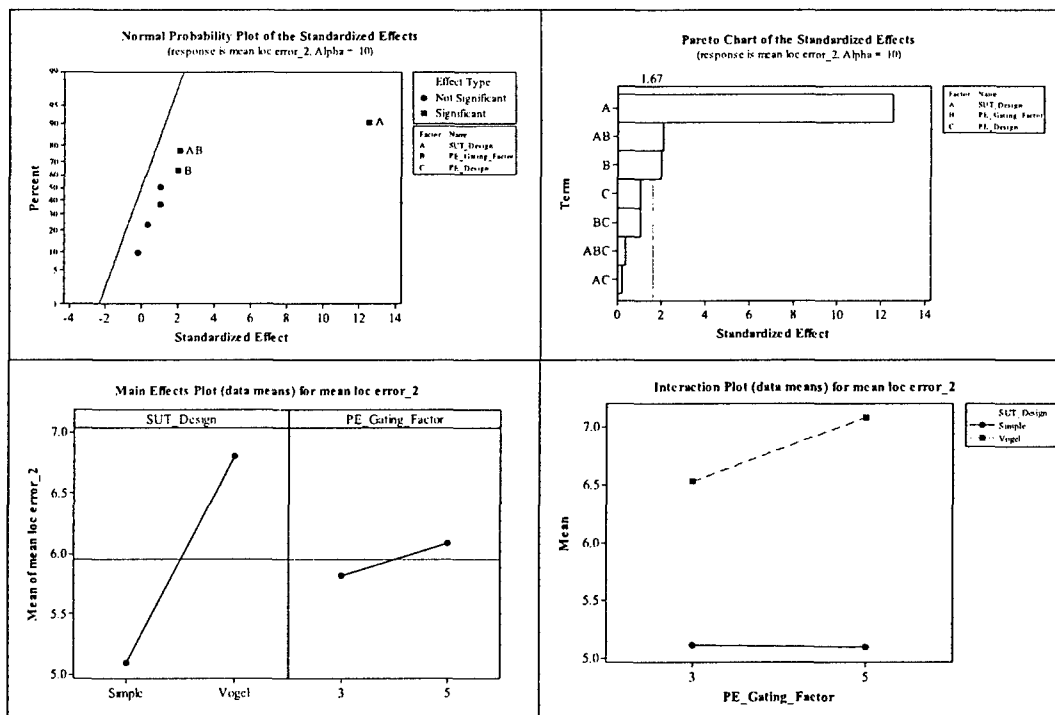
Track 2 to truth: Consistency



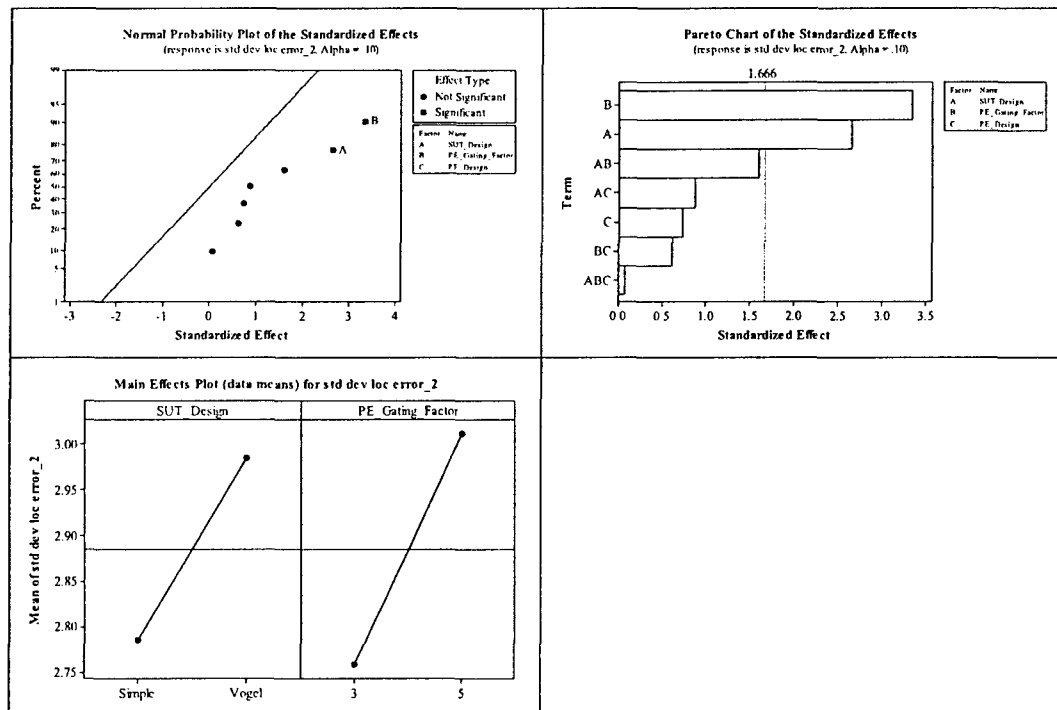
Track 2 to truth: Percentage of false tracks



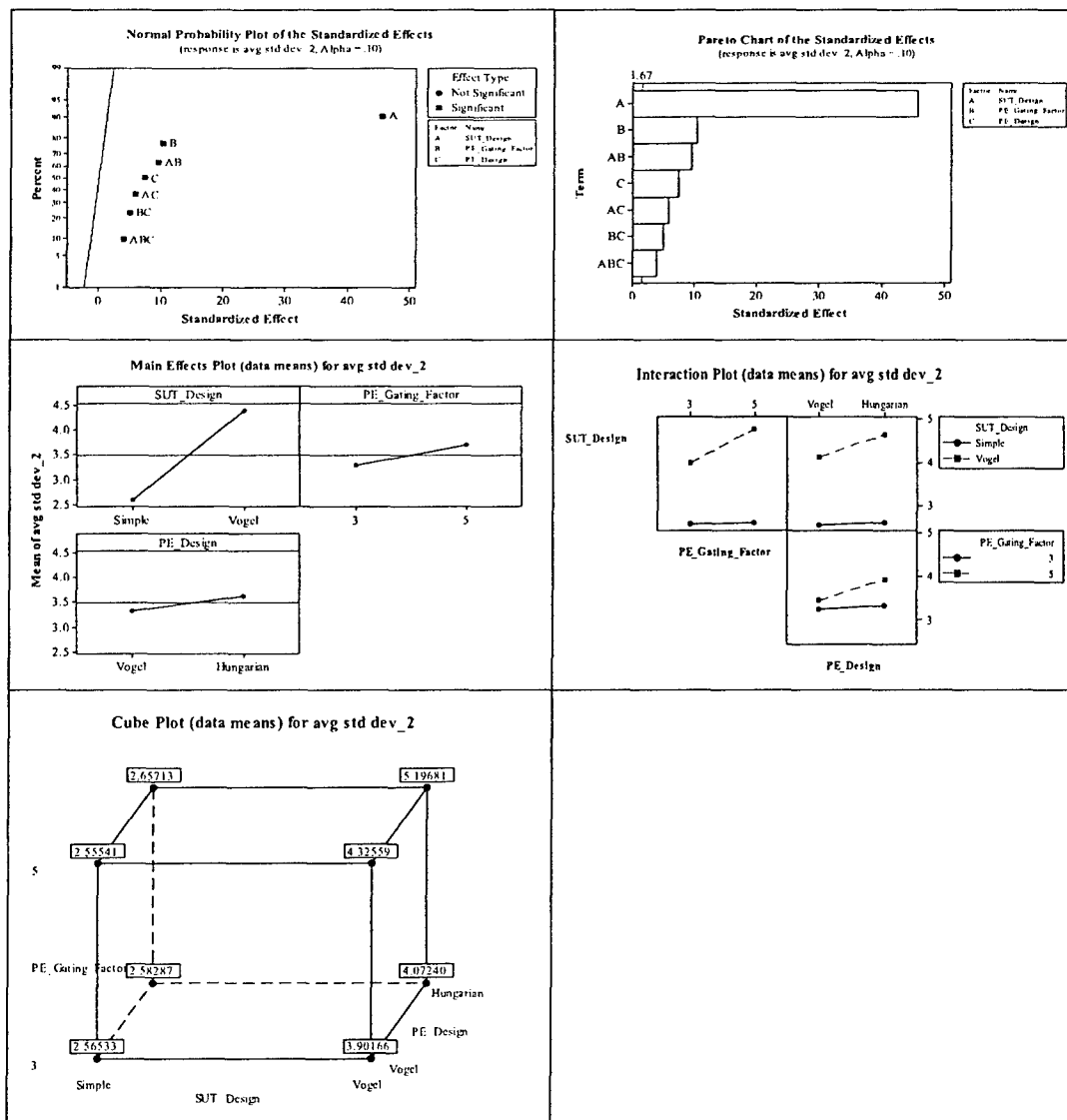
Track 2 to truth: Mean location error



Track 2 to truth: Average standard deviation of location error



Track 2 to truth: Average standard deviation

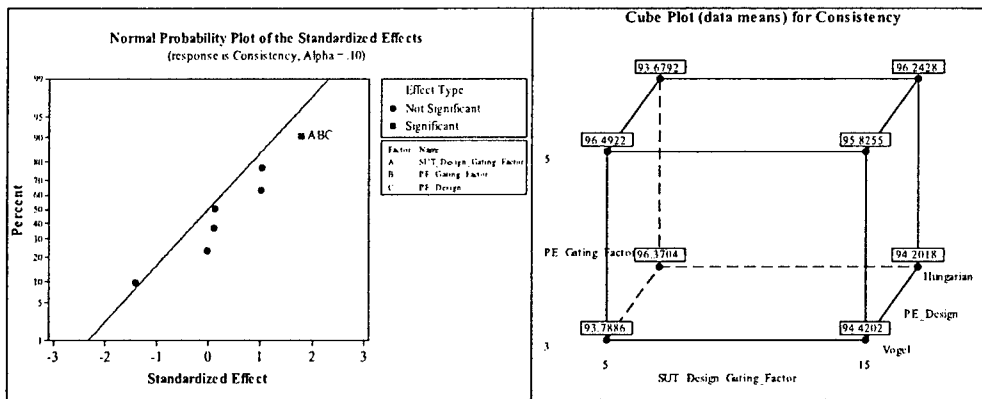


E. Tier 2 DOE Charts

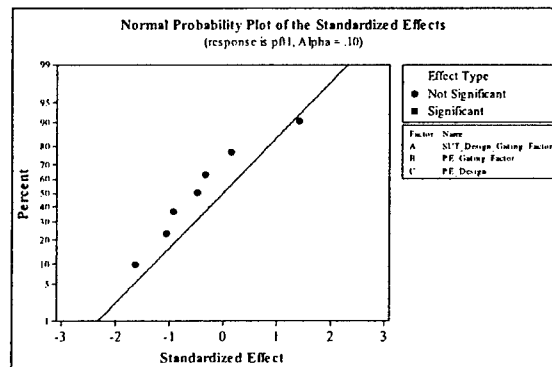
This section provides the Tier 2 DOE charts conducted in Section 9.1. The three factors SUT Design Gating Factor, PE Gating Factor and PE Design at two levels each are tested to find which of these factors affect the MOPs significantly. In Tier 2 we have three sensors on 2 platforms and they fuse data within and across platforms. So we have to analyze track-to-truth and track-to-track associations for each of the MOPs. The summary of the results is shown in Table 6. Here for each MOP we have the Normal Probability plot which summarizes the significant factors. Then for the significant factors we plot the main effects plot which tells us how the change in factor affects the MOP. For the significant interactions we plot the interaction plot which shows the effect of change in factor level combination on MOP.

After taking a look at the summary Table 6, we can say that none of the three factors SUT Design Gating Factor, PE Gating Factor and PE Design are significant. In this case only some of the two way and three way interactions are significant which suggests that fusing data across platforms reduces the discrepancies in the input data.

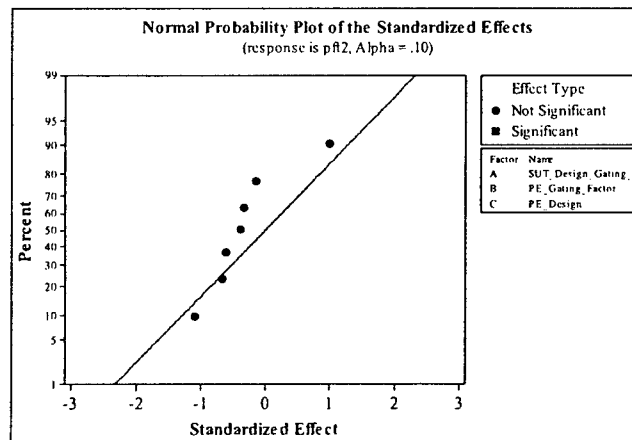
Track to track: Consistency



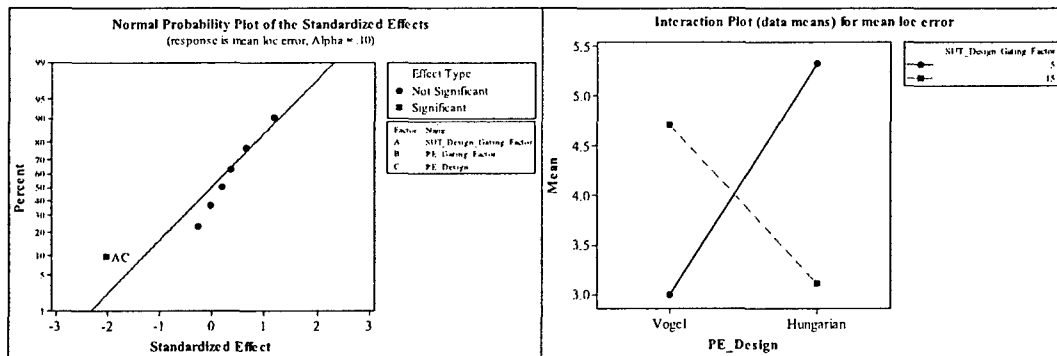
Track to track: Percentage of false tracks in track 1



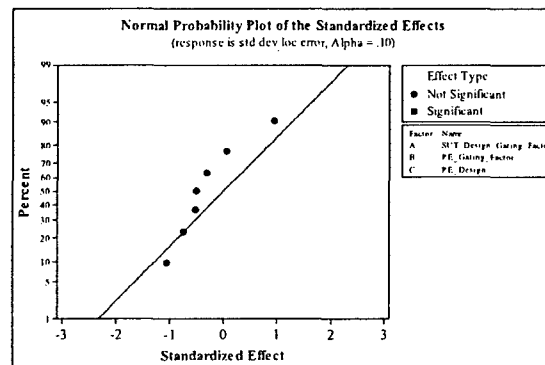
Track to track: Percentage of false tracks in track 2



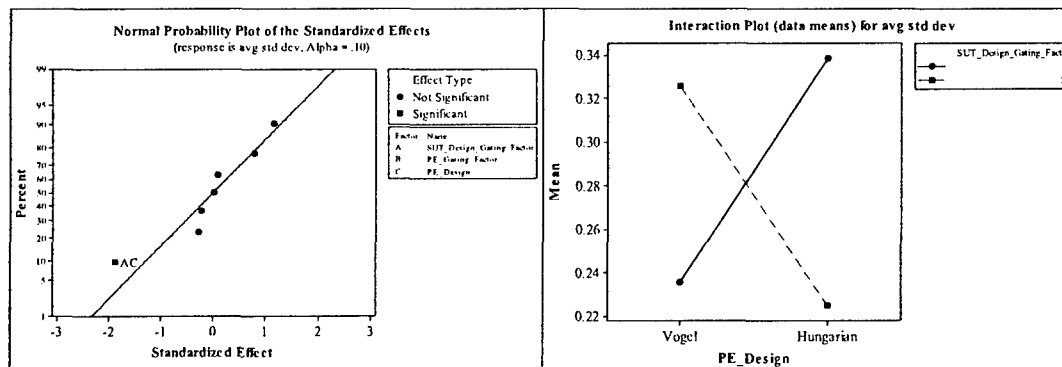
Track to track: Mean location error



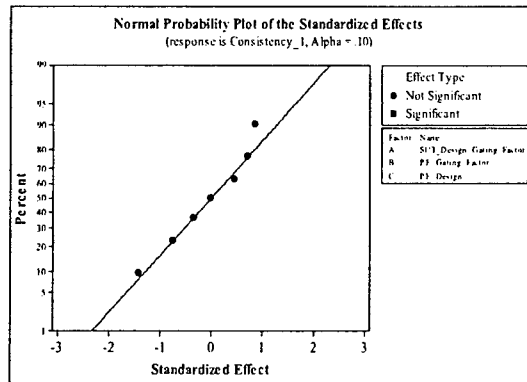
Track to track: Standard deviation of location error



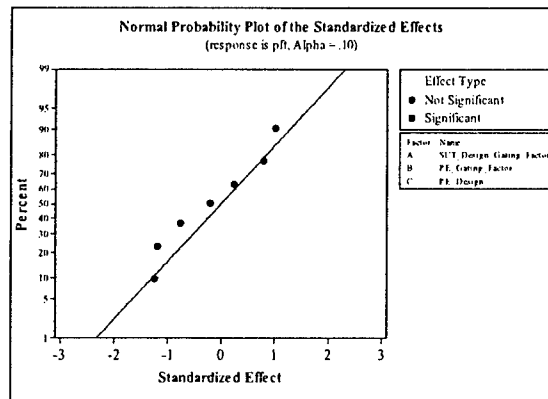
Track to track: Average standard deviation



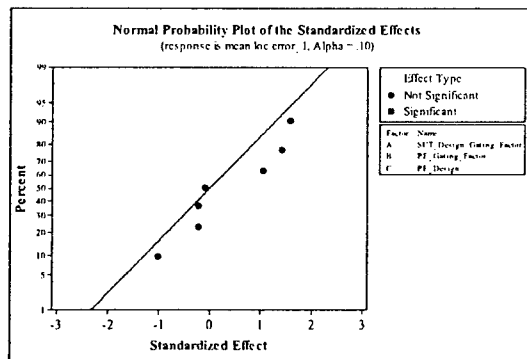
Track 1 to truth: Consistency



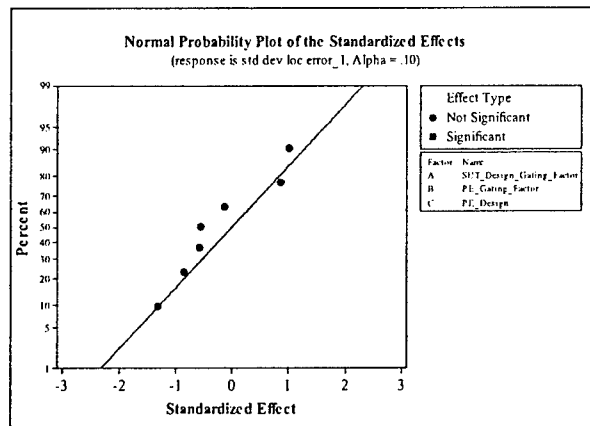
Track 1 to truth: Percentage of false tracks



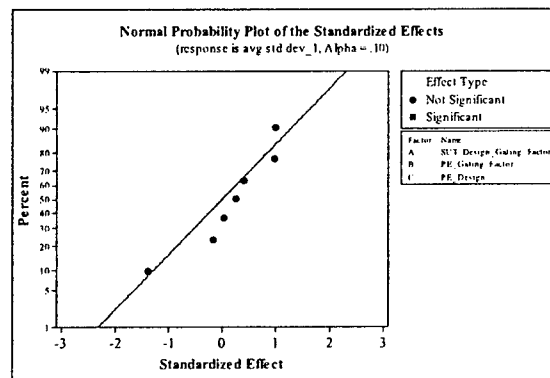
Track 1 to truth: Mean location error



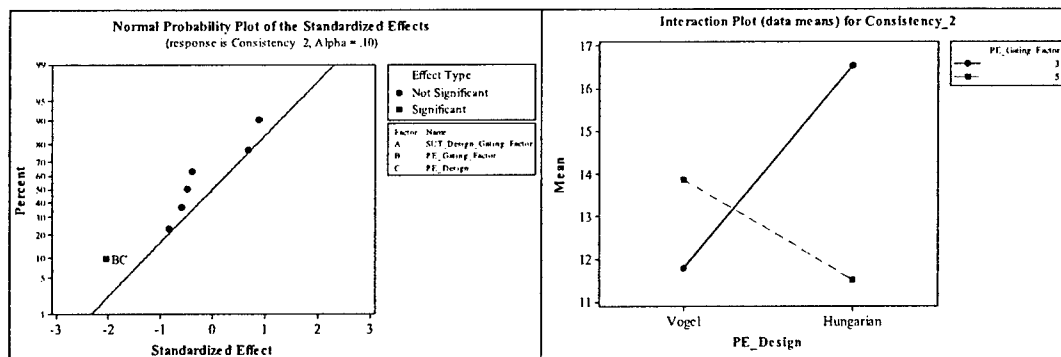
Track 1 to truth: Standard deviation of location error



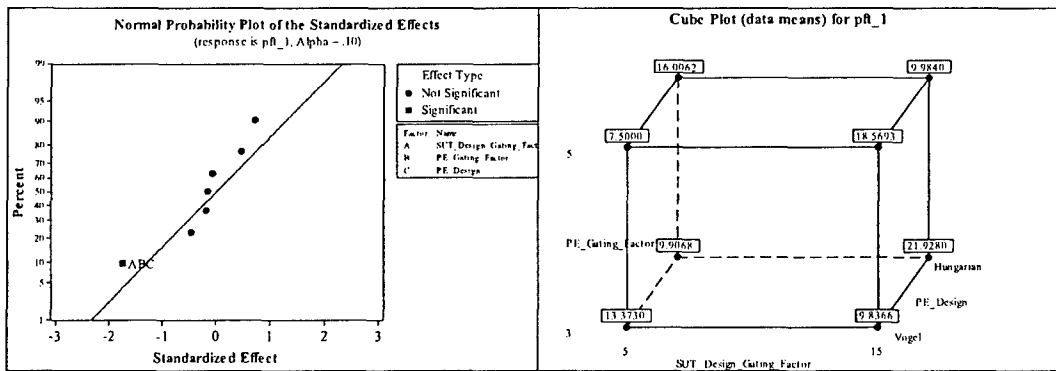
Track 1 to truth: Average standard deviation



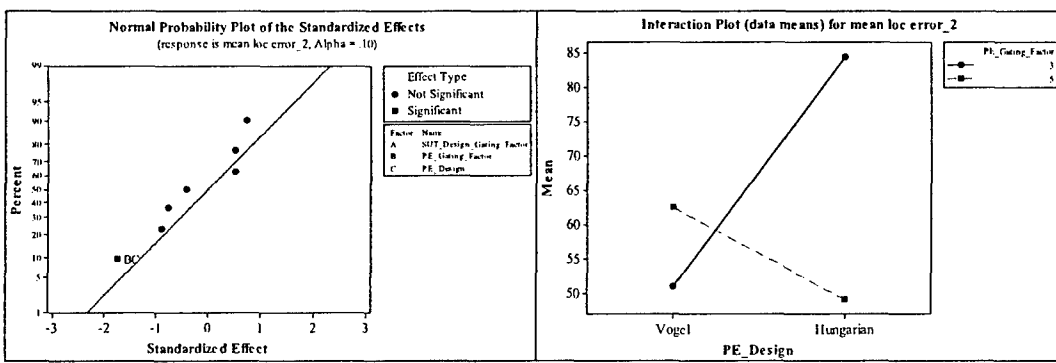
Track 2 to truth: Consistency



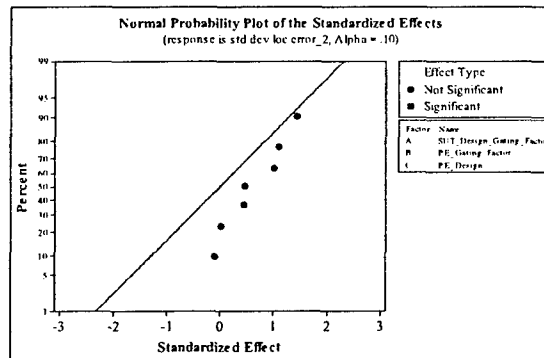
Track 2 to truth: Percentage of false tracks



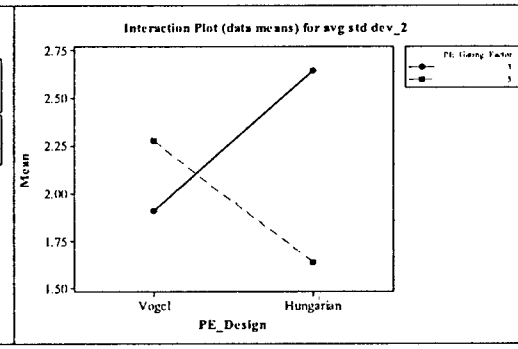
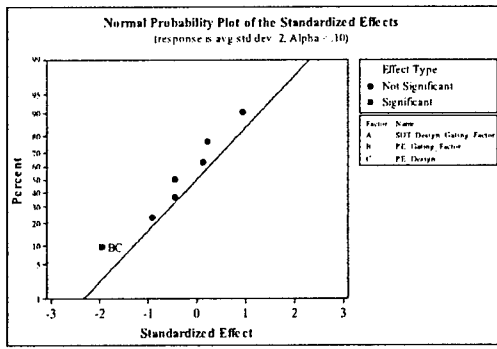
Track 2 to truth: Mean location error



Track 2 to truth: Standard deviation of location error

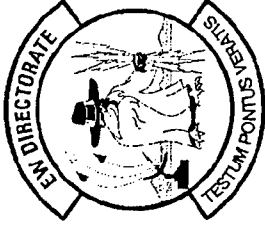


Track 2 to truth: Average standard deviation





PLAN



Experimental Design Approach

October 2005

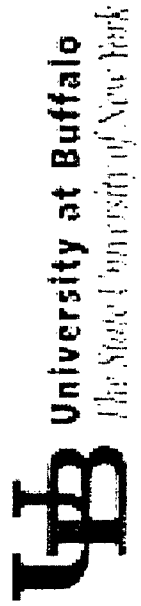
Kedar Sambhoos

kps6@bufflao.edu

Satyaki Ghosh Dastidar

sg44@buffalo.edu

Center for Multisource Information Fusion
University at Buffalo



PLAN

Introduction

Design of experiments will be conducted on Tier 2 of simulation (runSimulation2.m)

Number of Design of Experiments: 16

Track-To-Track: 6

Track-To-Truth: 5 (X 2)

Replication: 10

Simulation Runs: $(80 (2^3) * 16 = 1280)$

Measures of Performance

Track-To-Track

- Average Consistency (%)
Percentage of tracks for platform 1 that are not associated with tracks for platform 2 (%)
Percentage of tracks for platform 2 that are not associated with tracks for platform 1 (%)
- Average location error variance of associated tracks
(Location error mean, {km})
- Average location error variance of associated tracks
(Location error standard deviation, {km})
- The average number of standard deviations of error of the associated tracks (Average standard deviation)

Measures of Performance

Track-To-Truth (For each track to truth association)

- Percent Detection (Average Consistency)(%)
- Percentage of tracks that are false (%)
- Average location error variance of associated tracks (Location error mean, {km})
- Average location error variance of associated tracks (Location error standard deviation, {km})
- The average number of standard deviations of error of the associated tracks (Average standard deviation)

Factors for Design of Experiment

Scenario: Offensive Sweep 2vs6 Air-to-Air

PE factors:

Association

Vogel Approximation (PE 1)

Hungarian based association (PE 2)

Gate Size

3 and 5

System under test (SUT) :

Design

Gate only association (No Vogel or Hungarian association)

Association using Vogel approximation

Factors for Design of Experiment

| FACTORS | LEVELS | |
|---|---------------------|-----------------------------|
| Scenario | Fixed | |
| SUT Design | Gate only | Vogel |
| Performance evaluation gating factor | 3 | 5 |
| Performance evaluation association factor | Vogel Approximation | Hungarian based association |

Factors for Design of Experiment

| FACTORS | LEVELS | |
|---------|--------|----|
| | -1 | +1 |
| | -1 | +1 |
| | -1 | +1 |

Design of Experiment

[illegible]

Minitab based factorial design

Full Factorial Design

| | | | |
|----------|---|---------------------|------|
| Factors: | 3 | Base Design: | 3, 8 |
| Runs: | 8 | Replicates: | 10 |
| Blocks: | 1 | Center pts (total): | 0 |

All terms are free from aliasing.

Alias Structure

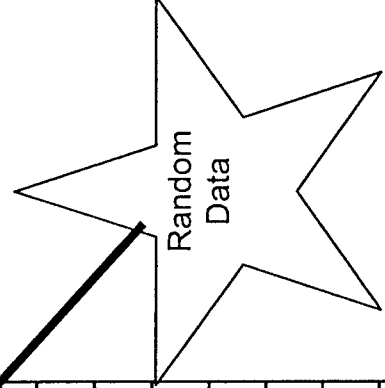
```
I
SUT_Design
PE_Gating_Factor
PE_Design
SUT_Design*PE_Gating_Factor
SUT_Design*PE_Design
PE_Gating_Factor*PE_Design
SUT_Design*PE_Gating_Factor*PE_Design
```

Run Order

| StdOrder | RunOrder | SUT Design | PE Gating Factor | PE Design | Track-to-Track Consistency |
|----------|----------|------------|------------------|-----------|----------------------------|
| 30 | 1 | Vogel | 5 | Hungarian | 67.347 |
| 1 | 2 | Vogel | 5 | Hungarian | 75.022 |
| 3 | 3 | Vogel | 5 | Vogel | 70.486 |
| 33 | 4 | Vogel | 5 | Vogel | 53.818 |
| 6 | 5 | Simple | 3 | Vogel | 70.601 |
| 32 | 6 | Vogel | 3 | Hungarian | 62.837 |
| 9 | 7 | Vogel | 5 | Vogel | 95.976 |
| 14 | 8 | Simple | 3 | Hungarian | 69.790 |
| 12 | 9 | Simple | 5 | Hungarian | 87.669 |
| 21 | 10 | Vogel | 3 | Vogel | 70.028 |
| 27 | 11 | Vogel | 3 | Vogel | 76.701 |
| 19 | 12 | Simple | 5 | Hungarian | 85.057 |
| 8 | 13 | Simple | 3 | Vogel | 89.101 |
| 16 | 14 | Simple | 5 | Hungarian | 89.032 |
| 18 | 15 | Simple | 5 | Vogel | 76.119 |
| 37 | 16 | Vogel | 5 | Vogel | 91.86f |
| ... | ... | ... | ... | ... | .. |

16 Minitab designs are formed corresponding to 16 MOP's

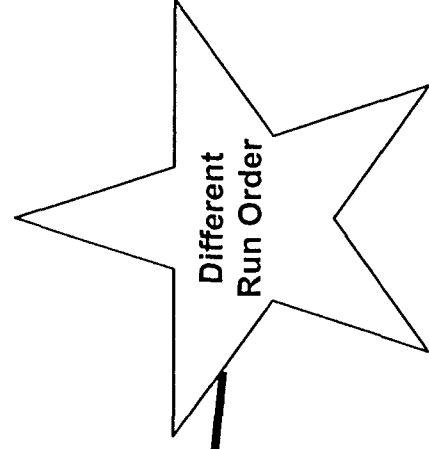
Each of the 16 Minitab designs has a different run order



Run Order

| 'stdOrder' | RunOrder | SUT Design | PE Gating Factor | PE Design | Track-to-Track PFT1* |
|------------|----------|------------|------------------|-----------|----------------------|
| 19 | 1 | Vogel | 5 | Hungarian | |
| 29 | 2 | Vogel | 5 | Hungarian | |
| 33 | 3 | Vogel | 5 | Vogel | |
| 38 | 4 | Vogel | 5 | Vogel | |
| 34 | 5 | Simple | 3 | Vogel | |
| 36 | 6 | Vogel | 3 | Hungarian | |
| 32 | 7 | Vogel | 5 | Vogel | |
| 27 | 8 | Simple | 3 | Hungarian | |
| 13 | 9 | Simple | 5 | Hungarian | |
| 8 | 10 | Vogel | 3 | Vogel | |
| 20 | 11 | Vogel | 3 | Vogel | |
| 5 | 12 | Simple | 5 | Hungarian | |
| 16 | 13 | Simple | 3 | Vogel | |
| 21 | 14 | Simple | 5 | Hungarian | |
| 25 | 15 | Simple | 5 | Vogel | |
| 3 | 16 | Vogel | 5 | Vogel | |
| ... | ... | ... | ... | ... | ... |

* Percentage of tracks for platform 1 that are not associated with tracks for platform 2



DOE Results*

Factorial Fit: Consistency versus SUT_Design, PE_Gating_Fa, PE_Design

Estimated Effects and Coefficients for Consistency (coded units)

| Term | Effect | Coef | SE Coef | T | P |
|------------------------------|--------|--------|---------|-------|-------|
| Constant | 75.321 | 1.188 | 63.41 | 0.000 | |
| SUT_Design | -0.508 | -0.254 | 1.188 | -0.21 | 0.831 |
| PE_Gating_Factor | -1.745 | -0.872 | 1.188 | -0.73 | 0.465 |
| PE_Design | 1.278 | 0.639 | 1.188 | 0.54 | 0.592 |
| SUT_Design*PE_Gating_Factor | 3.437 | 1.718 | 1.188 | 1.45 | 0.152 |
| SUT_Design*PE_Design | -0.138 | -0.069 | 1.188 | -0.06 | 0.954 |
| PE_Gating_Factor*PE_Design | -5.716 | -2.858 | 1.188 | -2.41 | 0.019 |
| SUT_Design*PE_Gating_Factor* | -2.168 | -1.084 | 1.188 | -0.91 | 0.365 |
| PE_Design | | | | | |

S = 10.6239 R-Sq = 11.76% R-Sq(adj) = 3.18%

Analysis of Variance for Consistency (coded units)

| Source | DF | Seq SS | Adj SS | Adj MS | F | P |
|--------------------|----|---------|---------|--------|------|-------|
| Main Effects | 3 | 98.72 | 98.72 | 32.91 | 0.29 | 0.831 |
| 2-Way Interactions | 3 | 889.97 | 889.97 | 296.66 | 2.63 | 0.057 |
| 3-Way Interactions | 1 | 93.97 | 93.97 | 93.97 | 0.83 | 0.365 |
| Residual Error | 72 | 8126.41 | 8126.41 | 112.87 | | |
| Pure Error | 72 | 8126.41 | 8126.41 | 112.87 | | |
| Total | 79 | 9209.08 | | | | |

* DOE results are based on random normal data created using Minitab with mean of 75 and standard deviation of 15

DOE Results*

Unusual Observations for Consistency

| Obs | StdOrder | Consistency | Fit | SE Fit | Residual | St Resid |
|-----|----------|-------------|--------|--------|----------|----------|
| 9 | 23 | 50.624 | 71.918 | 3.360 | -21.294 | -2.11R |
| 13 | 57 | 51.242 | 75.683 | 3.360 | -24.441 | -2.43R |
| 28 | 15 | 100.000 | 71.918 | 3.360 | 28.082 | 2.79R |
| 52 | 54 | 100.000 | 78.732 | 3.360 | 21.268 | 2.11R |

R denotes an observation with a large standardized residual.

Estimated Coefficients for Consistency using data in uncoded units

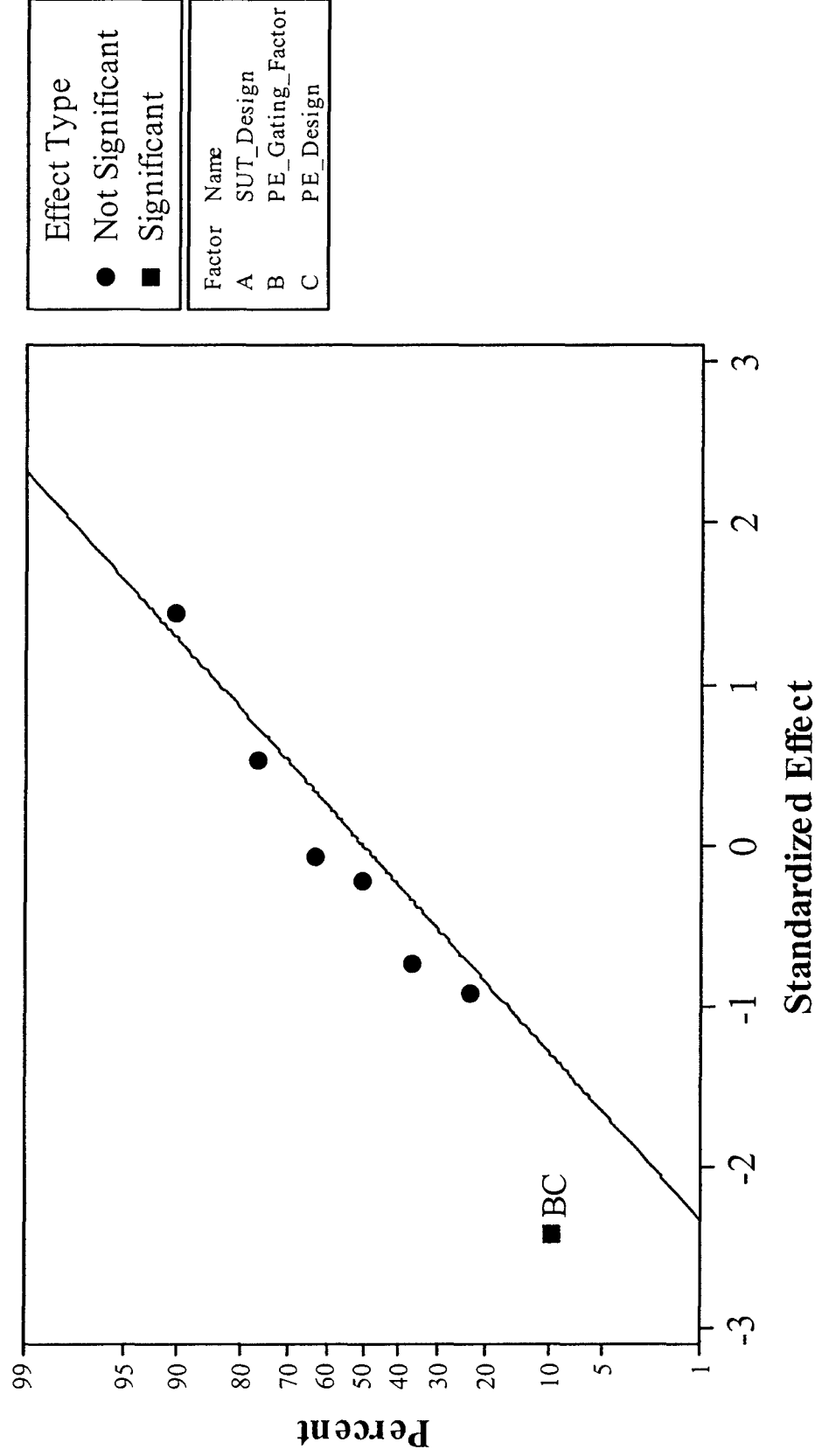
| Term | Coef |
|---|----------|
| Constant | 78.8102 |
| SUT_Design | -7.12734 |
| PE_Gating_Factor | -0.87239 |
| PE_Design | 12.0704 |
| SUT_Design*PE_Gating_Factor | 1.71839 |
| SUT_Design*PE_Design | 4.26606 |
| PE_Gating_Factor*PE_Design | -2.85780 |
| SUT_Design*PE_Gating_Factor* PE_Design | -1.08382 |

* DOE results are based on random normal data created using Minitab with mean of 75 and standard deviation of 15

DOE Results*

Normal Probability Plot of the Standardized Effects

(response is Consistency, Alpha = .10)

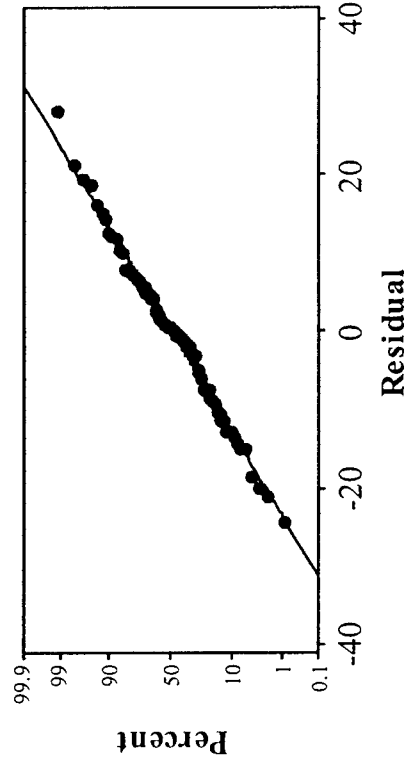


* DOE results are based on random normal data created using Minitab with mean of 75 and standard deviation of 15

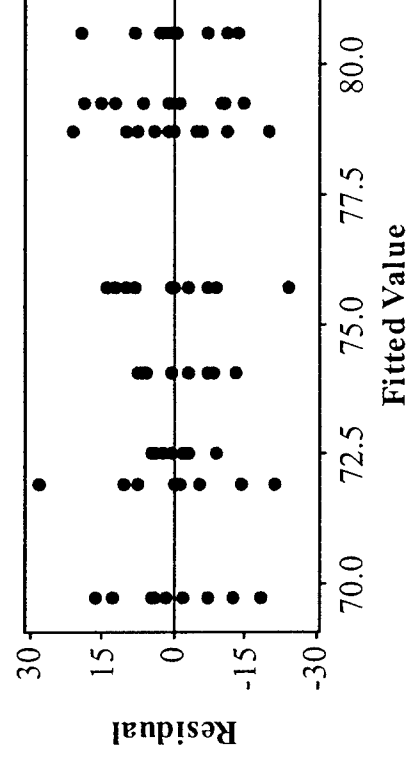
DOE Results*

Residual Plots for Consistency

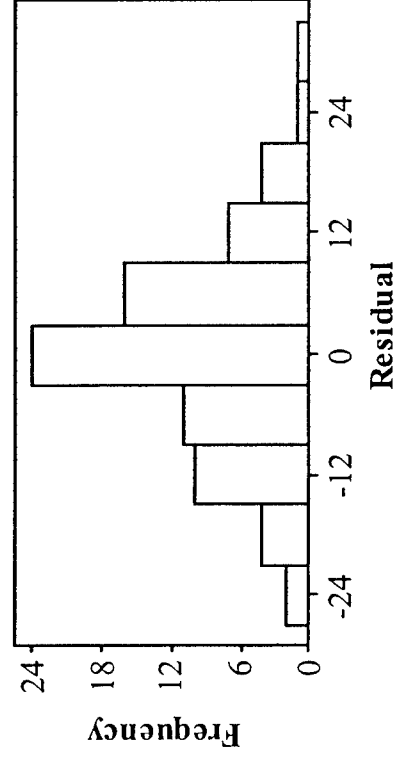
Normal Probability Plot of the Residuals



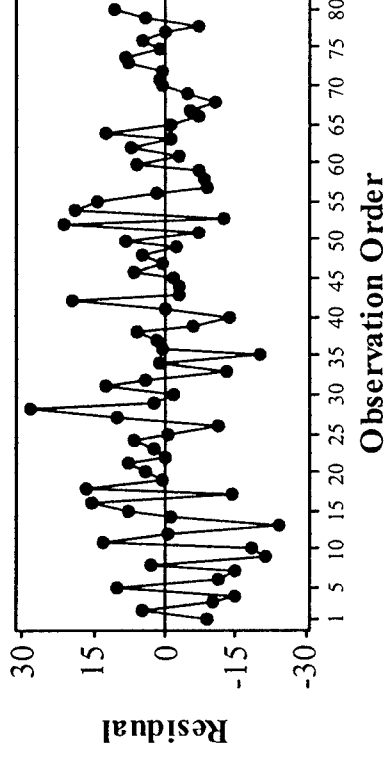
Residuals Versus the Fitted Values



Histogram of the Residuals



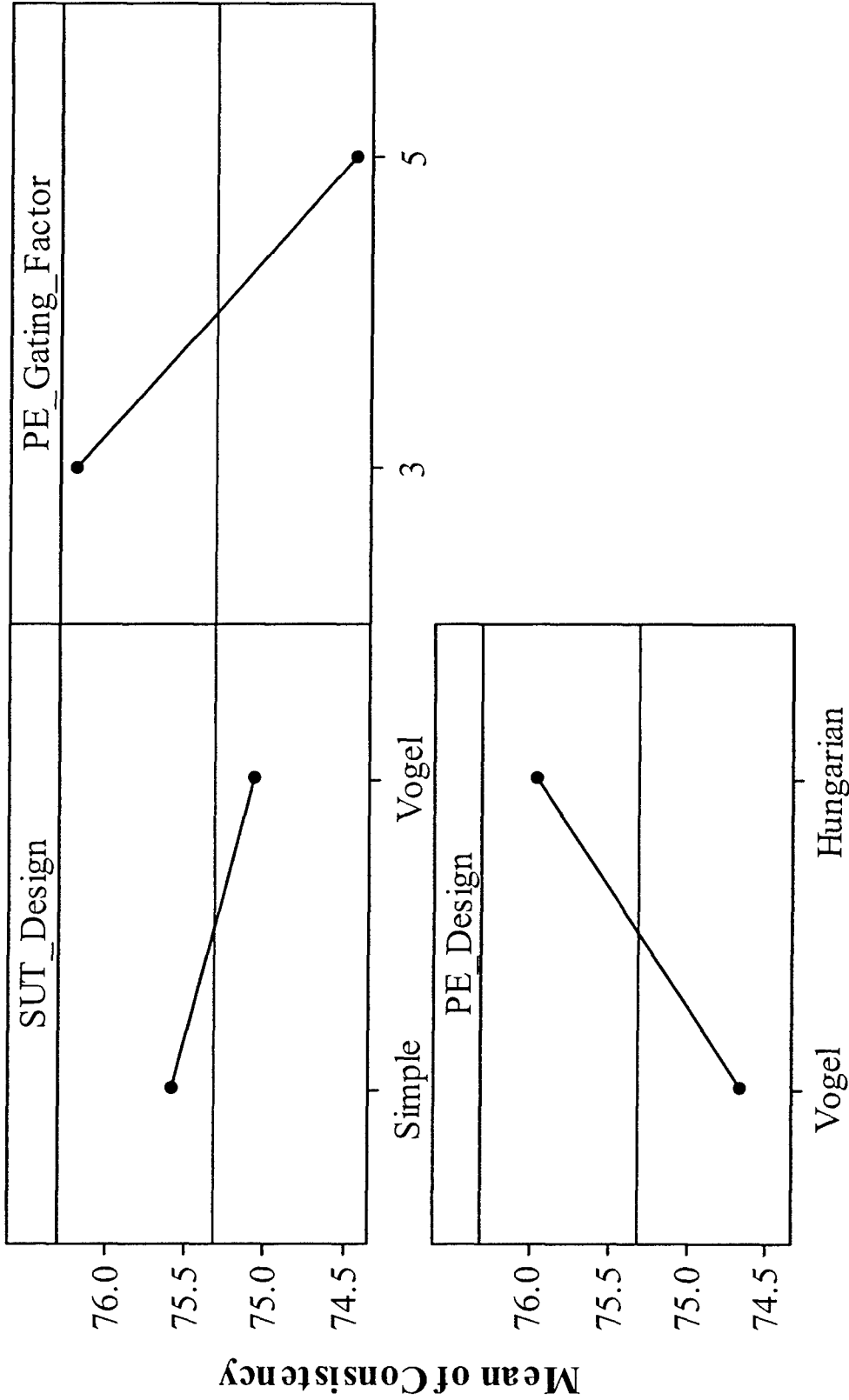
Residuals Versus the Order of the Data



* DOE results are based on random normal data created using Minitab with mean of 75 and standard deviation of 15

DOE Results*: Main Effect Plot

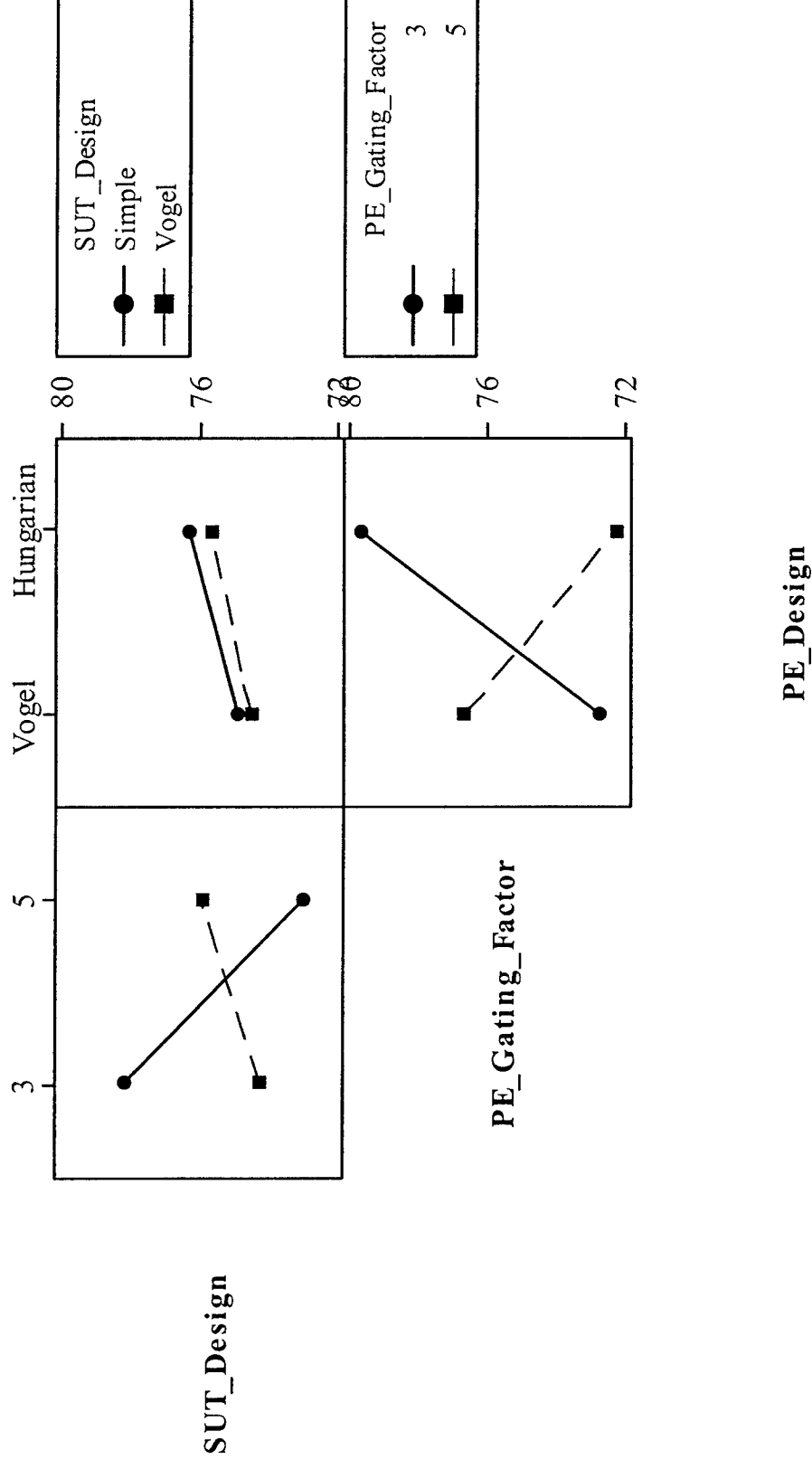
Main Effects Plot (data means) for Consistency



* DOE results are based on random normal data created using Minitab with mean of 75 and standard deviation of 15

DOE Results*: Interaction Plot

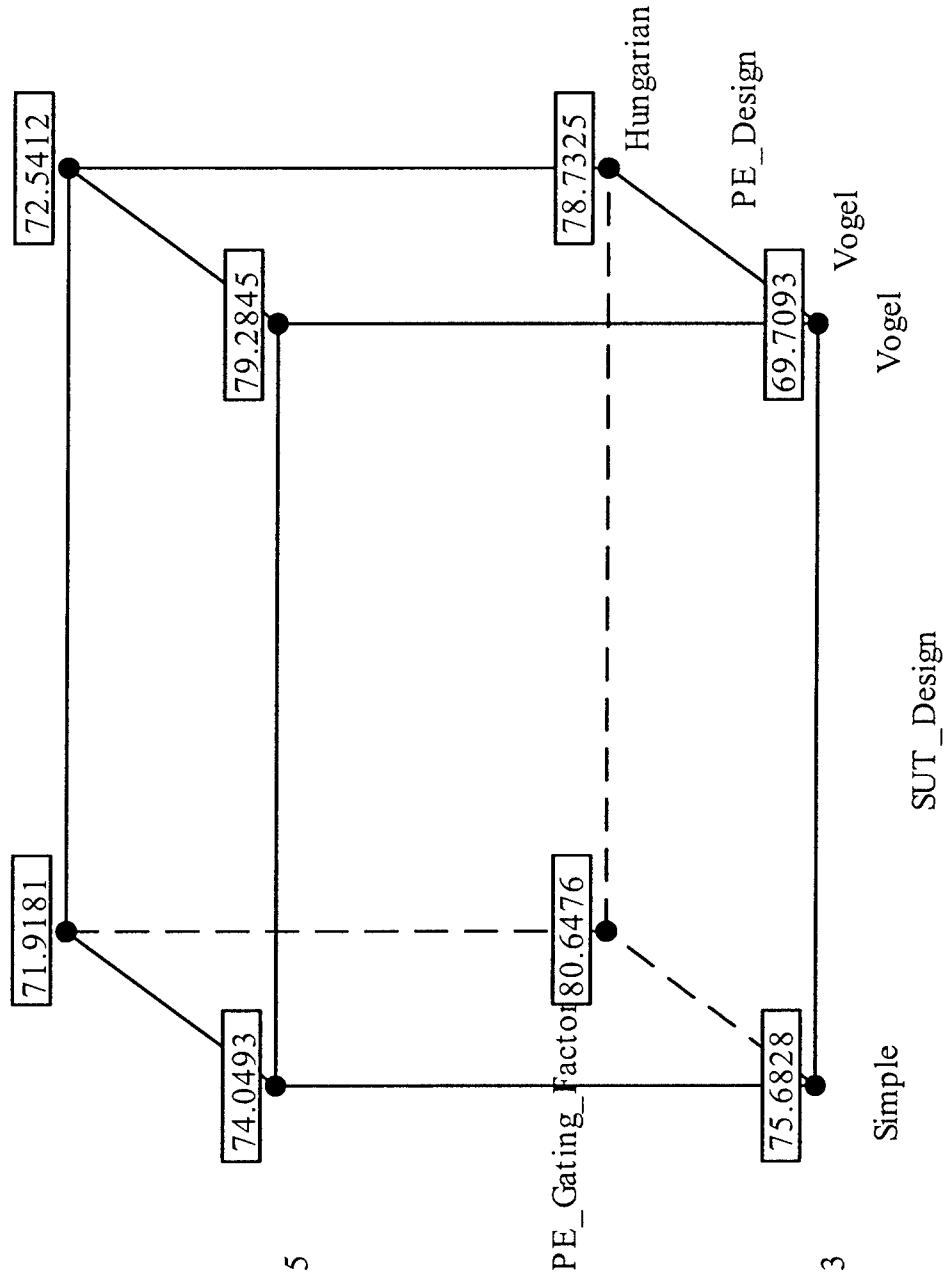
Interaction Plot (data means) for Consistency



* DOE results are based on random normal data created using Minitab with mean of 75 and standard deviation of 15

DOE Results*: Cube Plot

Cube Plot (data means) for Consistency



* DOE results are based on random normal data created using Minitab with mean of 75 and standard deviation of 15